



IMPLEMENTASI ALGORITMA BRANCH AND BOUND PADA 0-1 KNAPSACK PROBLEM UNTUK MENGOPTIMALKAN MUATAN BARANG

Arum Pratiwi ✉, Mulyono, Rochmad

Jurusan Matematika, FMIPA, Universitas Negeri Semarang, Indonesia
Gedung D7 lantai 1 Kampus Sekaran, Gunungpati, Semarang, 50229

Info Artikel

Sejarah Artikel:
Diterima Desember 2013
Disetujui Mei 2014
Dipublikasikan Nopember 2014

Keywords:
Integer Programming;
Knapsack Problem;
Branch and Bound Algorithm
Lingo 11.0.

Abstrak

Pemrograman bilangan bulat (*integer programming*) yang hanya mempunyai satu pembatas dikenal sebagai masalah knapsack (*knapsack problem*). Metode untuk menyelesaikan permasalahan tersebut adalah algoritma *Branch and Bound*. Algoritma *Branch and Bound* secara sistematis mengabaikan sekumpulan kandidat solusi yang tidak potensial menuju solusi optimal menggunakan estimasi batas atas dan bawah (*upper and lower estimated bounds*) dari kuantitas yang dioptimasi. Alat bantu yang dapat digunakan adalah *software Lingo 11.0*. Dalam tulisan ini akan dikaji bagaimana cara untuk mengoptimalkan muatan barang agar memperoleh keuntungan yang maksimal. Penyelesaian permasalahan *Knapsack 0-1* berdasarkan studi kasus di CV Pangestu Interaksi Semarang dengan menggunakan algoritma *Branch and Bound* diperoleh solusi optimal $x^*=(1,1,1,1,0,1)$ dan nilai optimal $z^*=1130000$ yang ditemukan pada submasalah 1, step kesembilan ($t=9$).

Abstract

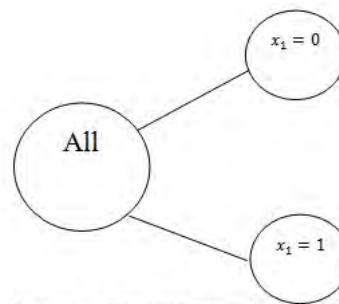
Integer Programming which has only one bound known as the Knapsack problem. The methods to solve these problems use Branch and Bound algorithm. Branch and Bound algorithm systematically ignore the solution candidate which not potential towards optimal solution used the estimated upper and lower bounds of optimized quantity. The software that can be used is Lingo 11.0. In this paper will be studied how to optimize a charge of items in order to obtain the maximum benefit. 0-1 Knapsack problem resolution based on cased studied in CV Pangestu Interaksi Semarang using the Branch and Bound algorithm obtained the optimal solution $x^*=(1,1,1,1,0,1)$ and optimal value of $z^*=1130000$ were found on subproblems 1, step nine ($t=9$).

Pendahuluan

Permasalahan optimasi merupakan permasalahan yang terjadi secara umum pada kehidupan manusia sehari-hari dalam usaha untuk memenuhi kebutuhannya. Optimasi adalah sebuah nilai yang memenuhi fungsi tujuan dan memenuhi segala persyaratan atau kendala yang terlibat (Taha, 1996). Bentuk optimasi dapat berupa meminimumkan biaya yang dikeluarkan atau memaksimumkan pendapatan yang ingin diperoleh. Model untuk merepresentasikan dan menyelesaikan persoalan tersebut adalah menggunakan pemrograman bilangan bulat (*integer programming*). Pemrograman bilangan bulat atau *Integer Programming* (IP) adalah bentuk lain dari Program Linear (LP) di mana asumsi divisibilitasnya melemah atau hilang sama sekali (Dimiyati & Tarliah, 1992).

Persoalan *integer programming* yang hanya mempunyai satu pembatas dikenal sebagai *knapsack problem* (Dimiyati & Tarliah, 1992). Permasalahan ini sering muncul dalam alokasi sumber daya dengan keuangan yang terbatas. Metode yang digunakan untuk mengoptimalkan muatan barang tersebut adalah algoritma *Branch and Bound*. Algoritma *Branch and Bound* pertama kali diusulkan oleh A.H. Land dan A.G. Doig pada tahun 1960 untuk *linear programming*. *Branch and Bound* adalah “suatu algoritma umum untuk pencarian solusi optimal dari berbagai masalah optimasi, khususnya untuk optimasi diskrit dan kombinatorial” (Suyanto, 2010).

Dalam bukunya yang berjudul *Operation Research*, Hiller & Lieberman (1980: 439) menjelaskan bahwa “setiap algoritma *Branch and Bound* memiliki tiga langkah dasar untuk setiap iterasi yaitu *branching*, *bounding* dan *fathoming* atau *pruning*. Fleksibilitas terletak pada bagaimana ketiga langkah ini dilakukan. Pembagian (*branching*) dilakukan dengan membagi keseluruhan himpunan solusi layak menjadi himpunan bagian yang lebih kecil. Diantara submasalah yang masih ada (belum dihilangkan), pilihlah satu submasalah yang dihasilkan paling akhir. Hentikan hubungan sampai ditemukan batas yang lebih besar. Cabangkan simpul untuk submasalah ini untuk menghasilkan submasalah yang baru dengan menetapkan variable pencabangan selanjutnya (*branching variable*) pada 0 atau 1.



Gambar 1. Pohon Solusi yang Dihasilkan dengan Pencabangan

Iterasi Pertama dari Algoritma *Branch and Bound*.

Gambar 1. Pohon Solusi yang Dihasilkan dengan Pencabangan Iterasi Pertama dari Algoritma *Branch and Bound*.

Pada Gambar 1 menggambarkan pembagian (*branching*) ini menjadi sub-sub masalah melalui sebuah pohon dengan cabang-cabang (busur) dari semua simpul (yang berhubungan dengan keseluruhan yang memiliki semua solusi yang layak) ke dua simpul yang berhubungan dengan kedua submasalah. Pohon ini yang akan terus “menumbuhkan cabang” iterasi demi iterasi., disebut pohon solusi (*enumeration tree*) untuk algoritma tersebut. Variabel yang digunakan untuk melakukan pencabangan ini pada setiap iterasi dengan menentukan nilai untuk variabel (seperti untuk x_1 di atas) disebut *branching variable*.

Pada setiap submasalah perlu mendapatkan batas (*bound*) mengenai seberapa jauh solusi layak terbaik dapat dicapai. Untuk submasalah baru, dapatkan *upper bound* dan *lower bound* kemudian bulatkan ke bawah nilai z nya untuk menghasilkan solusi yang optimal. Penghilangan (*fathoming*) sebagian dilakukan dengan membatasi (*bounding*) seberapa baik solusi terbaik dalam himpunan bagian dan kemudian membuang himpunan bagian apabila batasnya mengindikasikan bahwa himpunan bagian tersebut tidak mungkin berisi suatu solusi optimal untuk kasus awal. Suatu submasalah dihilangkan dari pertimbangan yang lebih jauh apabila memenuhi syarat berikut.

Uji 1 : batasnya $\leq z^*$.

Uji 2 : LP *relaxation*-nya tidak memiliki solusi layak.

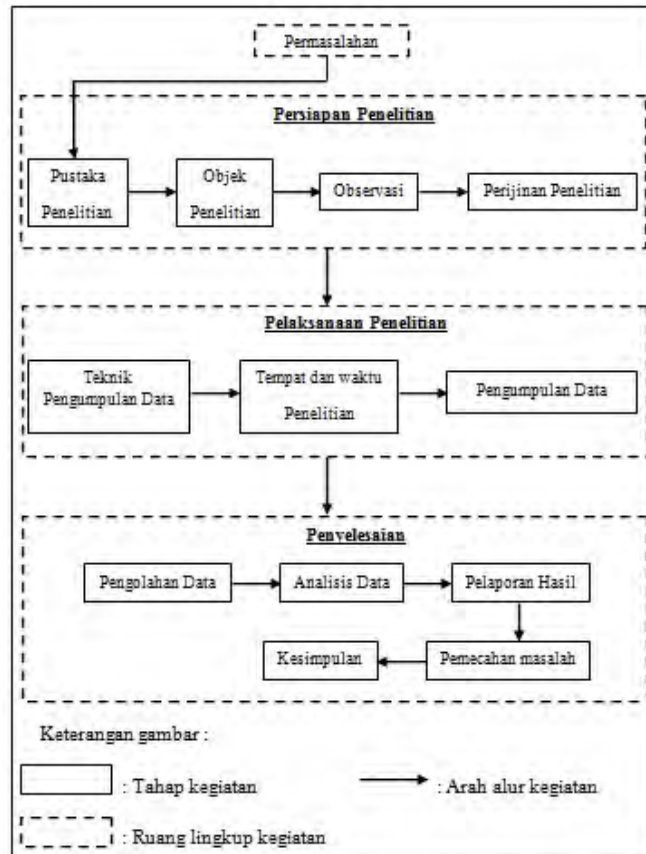
Uji 3 : solusi optimal untuk LP *relaxation*-nya adalah bilangan bulat.

Apabila solusi ini lebih baik daripada *incumbent* (solusi layak terbaik yang ditemukan sejauh itu), solusi ini akan menjadi *incumbent* baru dan uji 1 diterapkan kembali pada semua submasalah yang tidak dihilangkan dengan z^* baru yang lebih besar (z^* = nilai z untuk *incumbent* tertentu).

Dilakukan uji optimalitas apabila berhenti ketika tidak ada lagi submasalah yang tersisa, *incumbent* yang berlaku adalah optimal. Jika tidak, kembali lakukan iterasi lainnya.

Metode

Dalam melakukan penelitian harus memperhatikan prosedur dan langkah-langkah yang akan dilakukan untuk memulai penelitian sehingga dapat terarah dan terlaksana dengan baik dalam hal pelaporan penelitian. Langkah-langkah dasar yaitu: persiapan, penelitian, pelaksanaan penelitian dan pelaporan penelitian. Adapun alur yang menggambarkan kerja pada penelitian ini terdapat pada berikut:



Gambar 2. Diagram Alur Kerja.

Penyelesaian persoalan *integer programming* pada *knapsack problem* dapat diselesaikan dengan menghitung secara manual maupun dengan menggunakan bantuan *software* aplikasi jika dalam persoalan tersebut telah melibatkan banyak variabel. Dari latar belakang tersebut, maka penulis merumuskan beberapa permasalahan yaitu (1) Bagaimana solusi untuk mengoptimalkan muatan barang dengan algoritma *Branch and Bound* sehingga diperoleh keuntungan yang maksimum? (2) Bagaimana cara menggunakan aplikasi *software Lingo 11.0* untuk mengoptimalkan muatan barang dengan algoritma *Branch and Bound*?

Terdapat materi yang terkait dalam bidang teori graf dan program linear. Banyak aplikasi dalam graf dan program linear yang dapat diterapkan dalam kehidupan sehari-hari. Proses pengumpulan pustaka penelitian ini diambil dari berbagai sumber seperti buku-buku, artikel, jurnal dan literature lainnya. Dalam penelitian ini data yang yang diambil adalah data primer, yaitu data yang diperoleh dengan pengamatan secara langsung dari tempat penelitian, yaitu jenis barang yang diangkut ke dalam alat pengangkut atau media transportasi. Penelitian dilakukan di CV Pangestu Interaksi Semarang.

Data yang dikumpulkan pada penelitian ini adalah Data Persediaan Barang Bulan Agustus 2013 yang meliputi jenis barang, jumlah setiap jenis barang, berat setiap jenis barang, harga pokok setiap jenis barang serta harga jual setiap jenis barang. Data-data yang telah dikumpulkan dari lapangan tersebut kemudian diolah untuk memecahkan masalah penelitian.

dapat diangkut. Berdasarkan kapasitas alat angkut milik perusahaan tersebut, tentukan barang mana saja yang harus diangkut setiap minggunya untuk dipasarkan sehingga memperoleh keuntungan maksimum. Dalam hasil penelitian ini diberikan solusi pemecahannya menggunakan algoritma *Branch and Bound*. Diketahui setiap jenis barang memiliki berat, harga pokok dan harga jual yang berbeda-beda yang ditunjukkan dalam Tabel 1.

Tabel 1. Data Persediaan Barang Bulan Agustus 2013

| No | Nama Barang | Jumlah barang | @Berat barang | @Harga pokok | @ Harga jual |
|----|-----------------|---------------|---------------|---------------|---------------|
| 1. | Decglass | 10 box | 0,25 kg | Rp 25.000,00 | Rp 35.000,00 |
| 2. | Blood lakset | 30 box | 0,25 kg | Rp 17.500,00 | Rp 25.000,00 |
| 3. | Gluco Advantage | 8 box | 2,5 kg | Rp 240.000,00 | Rp 300.000,00 |
| 4. | Glucosa Repeed | 5 box | 2,5 kg | Rp 170.000,00 | Rp 220.000,00 |
| 5. | Golda A | 10 vial | 2,5 kg | Rp 80.000,00 | Rp 100.000,00 |
| 6. | Tabung HT | 15 buah | 1 kg | Rp 15.000,00 | Rp 20.000,00 |

Tabel 2. Data Pelengkap dari Tabel Utama

| No | Berat barang total (w) | Harga pokok total | Harga jual total | Keuntungan tiap jenis barang (p) | p/w |
|----|------------------------|-------------------|------------------|----------------------------------|--------------|
| 1. | 2,5 kg | Rp 250.000,00 | Rp 350.000,00 | Rp 100.000,00 | Rp 40.000,00 |
| 2. | 7,5 kg | Rp 525.000,00 | Rp 750.000,00 | Rp 225.000,00 | Rp 30.000,00 |
| 3. | 20 kg | Rp 1.920.000,00 | Rp 2.400.000,00 | Rp 480.000,00 | Rp 24.000,00 |
| 4. | 12,5 kg | Rp 850.000,00 | Rp 1.100.000,00 | Rp 250.000,00 | Rp 20.000,00 |
| 5. | 25 kg | Rp 800.000,00 | Rp 100.000,00 | Rp 200.000,00 | Rp 8.000,00 |
| 6. | 15 kg | Rp 225.000,00 | Rp 300.000,00 | Rp 75.000,00 | Rp 5.000,00 |

Hasil dan Pembahasan

Pada penelitian ini, yang akan dicari adalah keuntungan maksimum yang diperoleh perusahaan CV Pangestu Interaksi Semarang. Untuk itu perlu dilakukan pemilihan barang yang akan dipasarkan secara tepat sehingga bisa memperoleh keuntungan yang maksimal.

Permasalahan knapsack 0-1 yang dibahas dalam penelitian ini dapat diselesaikan dengan menggunakan algoritma *Branch and Bound*. Rincian permasalahan dijelaskan sebagai berikut.

CV Pangestu Interaksi adalah sebuah perusahaan yang bergerak di pemasaran produk dan alat-alat kesehatan di Semarang. Alat transportasi perusahaan tersebut mempunyai kapasitas alat angkut sebesar 60 kg. Untuk itu, dalam memasarkan produk dan alat-alatnya, perusahaan tersebut harus melakukan pemilihan barang, karena tidak semua barang

Berdasarkan penelitian Amaldi (2010), untuk menyelesaikan kasus pada penelitian di CV Pangestu Interaksi dapat disusun langkah-langkah perhitungan yang menggunakan algoritma *Branch and Bound* sebagai berikut. 1) Memodelkan permasalahan ke dalam bentuk *Integer Linear Programming*; 2) Mencari solusi optimal x dari C(KP); 3) Mencari nilai optimal z dari C(KP). 4) Melakukan pencabangan (*branching*) apabila telah memenuhi syarat; 5) Menentukan batas (*bounding*), yaitu batas atas (*upper bound*) dan batas bawah (*lower bound*). 6) Melakukan penghentian pencabangan (*fathoming*) apabila telah memenuhi syarat; 7) Menarik kesimpulan dari solusi optimal yang sudah diperoleh.

Berdasarkan langkah-langkah di atas, maka langkah pertama untuk menyelesaikan kasus pada penelitian ini adalah memodelkan permasalahan ke dalam bentuk *Integer Linear Programming*.

Misalkan x_j menyatakan keputusan perusahaan untuk mengangkut barang j , p_j adalah besarnya keuntungan setiap jenis barang yang diperoleh dari selisih harga jual dengan harga pokok, w_j adalah berat setiap jenis barang yang akan dipilih untuk dipasarkan dan z adalah total besar keuntungan yang diperoleh dari hasil pemilihan barang.

Bentuk *Integer Linear Programming* dari kasus di atas adalah sebagai berikut.

Fungsi tujuan:

$$\text{Maksimum dari } z = 100000x_1 + 225000x_2 + 480000x_3 + 250000x_4 + 200000x_5 + 75000x_6$$

dengan kendala

$$2,5x_1 + 7,5x_2 + 20x_3 + 12,5x_4 + 25x_5 + 15x_6 \leq 60$$

$$x_1, x_2, x_3, x_4, x_5, x_6 \in \{0,1\}$$

Langkah selanjutnya adalah mencari solusi optimal (\bar{x}) dari C(KP)

$$s = \min \left\{ j : \sum_{i=1}^j w_i > c \right\} \text{ Jelas } s=5, \text{ oleh sebab}$$

$$s = \min \left\{ j : \sum_{i=1}^j w_i > 60 \right\}$$

Selanjutnya \bar{x} dapat dicari berdasarkan persamaan berikut.

$$\bar{x}_j = 1, \text{ untuk } j = 1, \dots, s-1$$

$$\bar{x}_j = 0, \text{ untuk } j = s+1, \dots, n$$

$$\bar{x}_s = \frac{\bar{c}}{w_s}$$

dengan

$$\bar{c} = c - \sum_{j=1}^{s-1} w_j$$

Sehingga diperoleh solusi optimal (\bar{x})

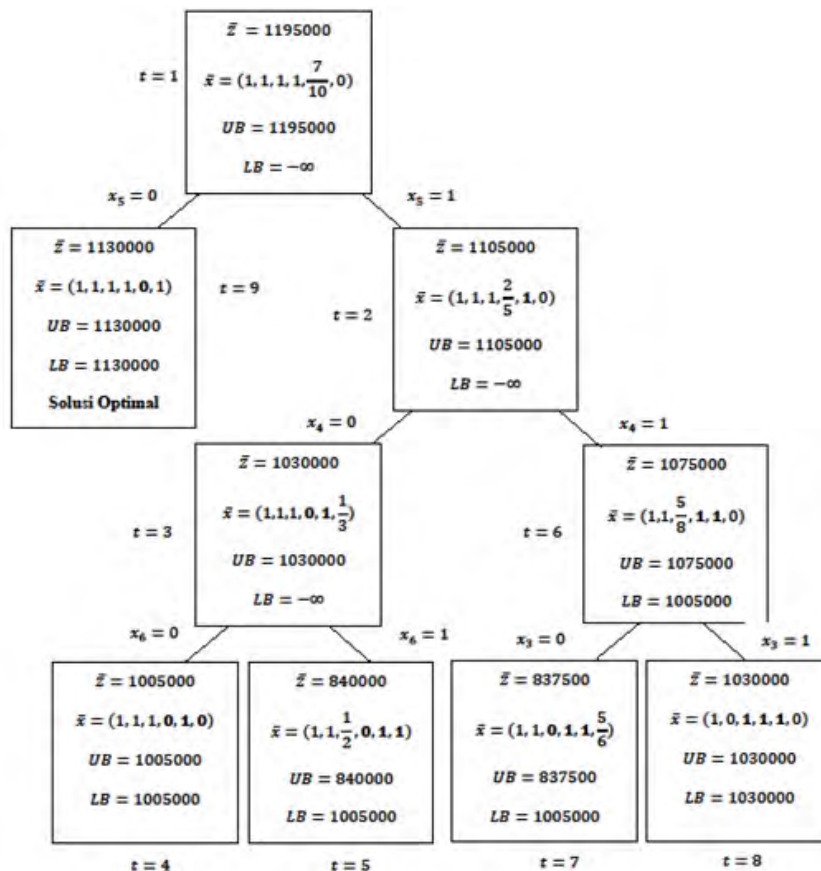
$$\bar{x}_1 = 1, \bar{x}_2 = 1, \bar{x}_3 = 1, \bar{x}_4 = 1, \bar{x}_5 = \frac{7}{10}, \bar{x}_6 = 0$$

Nilai optimal \bar{z} dicari dengan berdasarkan persamaan berikut.

$$z(C(KP)) = \sum_{j=1}^{s-1} p_j + \left[\bar{c} \frac{p_s}{w_s} \right]$$

$$z(C(KP)) = 1195000$$

Pohon solusi akhir ditunjukkan dalam gambar 3.



Gambar 3. Pohon Solusi Akhir ($t = 9$).

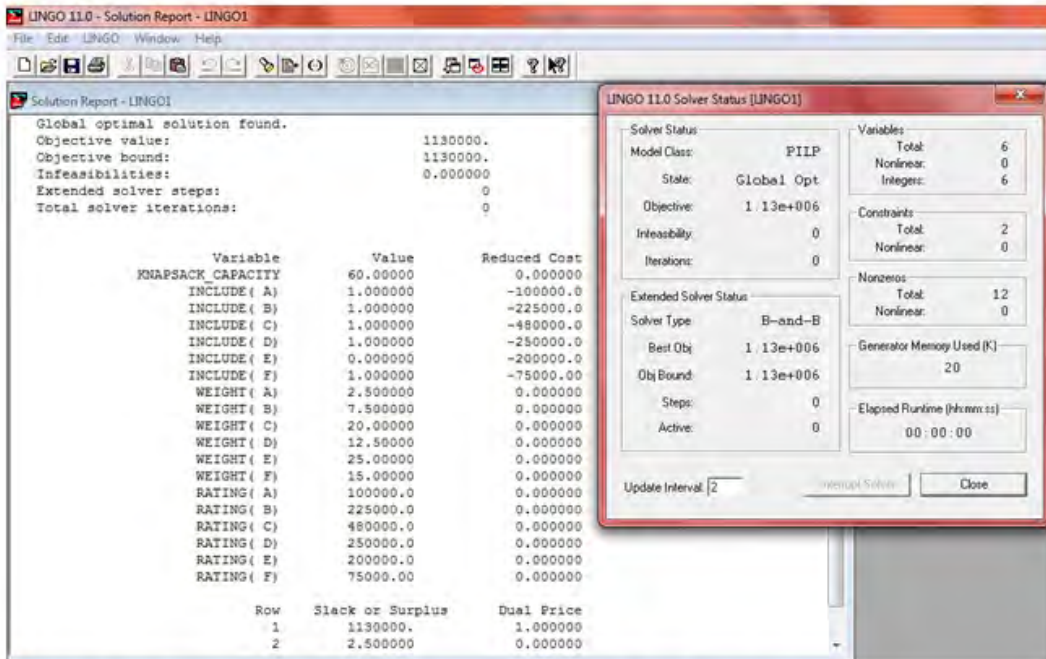
Berdasarkan penelitian Amaldi (2010) diperoleh beberapa penjelasan mengenai pohon solusi akhir berikut.

Indeks t menunjukkan urutan dimana submasalah dipecahkan. Saat nilai solusi optimal \bar{z} berupa pecahan, batas atas (*upper bound*) untuk submasalah dapat dibulatkan menuju $\lfloor \bar{z} \rfloor$. Batas bawah (*lower bound*) tidak dihitung untuk setiap submasalah namun akan berganti baru ketika sudah ditemukan submasalah dengan solusi yang optimal. Pada submasalah 5 ini dilakukan *fathoming* karena sudah memenuhi syarat 3. Pada submasalah 6, 7 ini dilakukan *fathoming* karena memenuhi syarat 1. Pada submasalah 1, 8 ini pencabangan dihentikan karena sudah memenuhi syarat 1 dan 3. Solusi optimal akhir ditemukan pada node 1 ($t=9$) dengan $x^*=(1,1,1,1,0,1)$ dan nilai optimal $z^*=1130000$.

Dalam *software Lingo.11* memuat metode *Branch and Bound*, untuk itu program ini bisa digunakan sebagai alat bantu untuk menyelesaikan permasalahan yang berkaitan tentang algoritma *Branch and Bound* salah satu contohnya adalah masalah *knapsack 0-1*. Berdasarkan hasil penelitian, kasus yang dibahas merupakan permasalahan *knapsack 0-1* sehingga kasus ini juga dapat diselesaikan menggunakan *software* ini. Berikut adalah hasil perhitungan dengan menggunakan bantuan *software Lingo 11.0*.

Pada *software Lingo.11.0*, tabel solver status menyatakan bahwa penyelesaian perhitungan *knapsack* ini menggunakan solver tipe *B-and-B* yang berarti *Branch and Bound*. Sedangkan pada *form solution report* diperoleh *objective value* (nilai keuntungan optimal). Kemudian solusi optimal yang diperoleh dapat dilihat dari kolom value yang menyatakan keputusan tiap-tiap variabel atau jenis barang. Jika nilai pada kolom value adalah 1 maka jenis barang tersebut diangkut sedangkan jika nilainya adalah 0 maka jenis barang tersebut tidak diangkut.

Berdasarkan penelitian oleh Amaldi (2010) dapat disusun langkah-langkah penyelesaian masalah *knapsack* menggunakan algoritma *Branch and Bound*. Dalam jurnal tersebut permasalahan *knapsack* yang diselesaikan adalah masalah *knapsack 0-1*. Kasus yang dibahas adalah bagaimana menentukan pemilihan investasi Bank ke setiap perusahaan supaya memperoleh total pendapatan maksimum. Kasus ini serupa dengan studi kasus yang dilakukan peneliti di CV Pangestu Interaksi Semarang, yaitu menentukan pemilihan pengangkutan barang untuk dipasarkan supaya memperoleh keuntungan penjualan yang maksimum.



Gambar 4. Output Penyelesaian Masalah dan Solusi Akhir dengan *Lingo 11.0*.

Pada studi kasus di CV Pangestu Interaksi Semarang, hasil perhitungan berdasarkan data persediaan bulan Agustus 2013 diperoleh solusi optimal $x^*=(1,1,1,1,0,1)$. Berdasarkan solusi optimal x^* , jenis barang yang terpilih adalah *decglass*, *blood lakset*, *Gluc Advantage*, *Glucosa Repeed*, dan Tabung HT. Apabila barang tersebut dipasarkan akan menghasilkan keuntungan penjualan maksimal sebesar Rp 1.130.000,00.

Simpulan

Penyelesaian masalah *Knapsack 0-1* dengan menggunakan algoritma *Branch and Bound* untuk studi kasus pada bulan Agustus 2013 di CV Pangestu Interaksi diperoleh total keuntungan Rp 1.130.000,00 dengan jenis barang yang diangkut adalah *decglass*, *blood lakset*, *Gluc Advantage*, *Glucosa Repeed*, dan Tabung HT. Cara menggunakan *software Lingo 11.0* untuk menyelesaikan masalah *Knapsack 0-1* dengan algoritma *Branch and Bound* pada kasus di CV Pangestu Interaksi adalah dengan menginputkan kode program ke dalam *form* perintah *software Lingo 11.0*. Dalam kode program tersebut diperlukan nilai dari masing-masing koefisien x pada fungsi tujuan dan nilai koefisien x pada kendala yang termuat dalam model matematika untuk kasus pada CV Pangestu Interaksi ini.

Daftar Pustaka

- Amaldi, E. 2010. *Foundation of Operation Research*. Politecnico di Milano.
- Dimiyati, A & Tarliah, T. 1992. *Operation Research. Model-model Pengambilan Keputusan*. Bandung: Sinar Baru Algensindo.
- Suyanto. 2010. *Algoritma Optimasi (Deterministik atau Probabilitik)*. Yogyakarta: Graha Ilmu.
- Taha, A.H. 1996. *Riset Optimasi*. Jakarta: Binarupa Aksara.