

OPTIMASI CUTTING STOCK PADA INDUSTRI PEMOTONGAN KERTAS DENGAN MENGUNAKAN METODE *INTEGER LINEAR PROGRAMMING* (Studi Kasus di Bhinneka – Semarang)

Denny Nurkertamanda, Singgih Saptadi, Adhika Permanasari
Teknik Industri Universitas Diponegoro

Abstract

In paper cutting industry, cutting stock problem (CSP) is a problem about how to cutting paper depends on quantity and specify of the demand. CSP related with dimension of pieces and rectangle which is use. In this research, we use one type dimension of rectangle and six type dimension of pieces and cutting all paper by two stage guillotine pattern. The major focus of this research is to formulate the paper cutting problem using integer linear programming.

Cutting large objects into small pieces can be found in many industries. Inevitably, the cutting processes produce trim loss. On the rectangle we can put some different dimension of pieces then we can make certain pattern. The modification pattern have to produce minimum trim loss. Thus to develop optimal cutting pattern to reduce trim loss is the main purpose of this research. To reach that, we use branch and bound algorithm then continued with sensitivity analysis.

From the research, we get optimum patten of paper cutting and quantity production for that pattern. Decision for quantity production depends on average demand every day. Beside that, we also give some alternative rules of production system which can take by the company.

Keywords : *Cutting stock problem, two stage guillotine pattern, branch and bound algorithm*

BAB I PENDAHULUAN

1.1 Latar Belakang

Kertas merupakan salah satu sarana yang digunakan di dalam komunikasi, sebagai media penyampaian informasi serta penyimpan informasi. Tidak dapat dipungkiri bahwa kertas memegang peranan yang cukup penting dalam kehidupan masyarakat.

Peran penting tersebut mengakibatkan kertas harus memiliki sifat *adjustable*, yaitu dapat disesuaikan dengan kebutuhan dari konsumennya khususnya mengenai ukuran atau dimensi dari kertas tersebut. Tidak semua konsumen memiliki kebutuhan kertas dengan dimensi yang sama dan tidak selamanya seorang konsumen memerlukan kertas dengan dimensi yang sama. Oleh karena itu, dimensi kertas harus sesuai dengan kebutuhan konsumen.

Bahan baku mengambil peranan yang sangat penting di dalam efisiensi produksi. Optimalisasi bahan baku perlu dilakukan guna mendapatkan tingkat

efisiensi produksi yang tinggi, yaitu sisa pemotongan kertas yang seminimum mungkin. Bahan baku yang digunakan oleh CV. Bhinneka berupa kertas yang nantinya akan dipotong sesuai dengan pola pemotongan yang telah ditentukan. Pola pemotongan yang dilakukan selama ini adalah dalam satu *rectangle* terdapat beberapa *pieces* dengan ukuran yang sama. Dari studi yang telah dilakukan diketahui bahwa ternyata dengan pola pemotongan seperti itu terdapat banyak sisa kertas yang terbuang. Diperlukan pola pemotongan baru yang dapat meminimasi sisa kertas tersebut.

Cutting stock problems merupakan permasalahan optimasi dalam pengkombinasian, sehingga dapat ditentukan solusi dari beberapa solusi yang mungkin, yang memenuhi fungsi pembatas yang ada. Solusi yang ditawarkan adalah dengan mengkombinasikan beberapa *pieces* dengan ukuran berbeda ke dalam persegi empat (bahan baku) sehingga didapatkan sisa kertas seminimal mungkin. Namun jika solusi tersebut dijalankan maka muncul masalah baru mengenai jumlah persegi

empat (bahan baku) yang harus diproduksi, mengingat permintaan dari setiap *pieces* tidaklah sama.

1.2 Perumusan Masalah

Permasalahan yang akan diangkat adalah mengenai pola pemotongan kertas seperti apa yang menghasilkan sisa area kertas minimum.

1.3 Tujuan Penelitian

Tujuan dari penelitian ini adalah sebagai berikut :

1. Mendapatkan rumusan komposisi *pieces* yang optimum untuk permasalahan CV. Bhinneka dengan menggunakan *Integer Linear Programming* serta memperoleh pola pemotongannya.
2. Mendapatkan jumlah produksi yang tepat untuk setiap pola pemotongan, yang sesuai dengan permasalahan pada CV. Bhinneka, dengan menggunakan *Integer Linear Programming*.
3. Memberikan masukan pada CV. Bhinneka mengenai alternatif pola pemotongan kertas yang mempertimbangkan optimasi dari pemotongan kertas serta jumlah produksi dari setiap pola pemotongan.
4. Menentukan sistem pemenuhan permintaan yang menunjang implementasi dari pola pemotongan yang baru.

1.4 Pembatasan Masalah

Dalam penelitian ini terdapat beberapa pembatasan yang digunakan, yaitu :

1. Pemotongan dilakukan secara *two stage guillotine pattern*.
2. Proses pemotongan dilakukan dengan mesin yang hanya mempunyai satu pisau potong, sehingga tidak dapat melakukan pemotongan secara paralel.
3. Ukuran *rectangle* yang digunakan adalah 100 cm x 65 cm kertas HVS 80 gr.
4. Ukuran kertas *pieces* yang digunakan ada 6 jenis, yaitu folio, A₃, A₄, A₄S, A₅, dan F₄.

5. Data permintaan yang digunakan adalah data bulan Juni 2006.
6. Hanya memperhatikan atau membahas dari segi sisa kertas yang dihasilkan saja.

BAB II TINJAUAN PUSTAKA

2.1 Karakteristik Pemotongan Bahan (*Cutting Stock*)

Karakteristik pemotongan bahan adalah :

1. Terdapat bahan baku yang berbentuk persegi empat (selanjutnya disebut *rectangle*) yang mempunyai ukuran tertentu.
2. Terdapat m jenis potongan yang dihasilkan (yang selanjutnya disebut dengan *pieces*) yang masing-masing berukuran $p_i \times l_i$ ($i = 1..m$) dengan jumlah permintaan n tertentu.
3. Setiap potong mempunyai nilai tertentu (v_i) yang bisa berupa keuntungan yang diperoleh atau berupa ukuran luas dalam upaya meminimasi sisa bahan baku.
4. Berusaha membentuk suatu *layout* potong yang meminimumkan fungsi tujuan yang melekat pada setiap potong yang ada.

2.2 Jenis-Jenis Pemotongan Bahan (*Cutting Stock*)

1. Berdasarkan jumlah dimensi yang dipertimbangkan
 - a. *One dimensional*
 - b. *Two dimensional*
 - c. *Three dimensional*
2. Berdasarkan jenis penugasan
 - a. *Big material to small pieces*
 - b. *Small pieces to big material*
3. Berdasarkan pada jumlah bahan yang dipertimbangkan
 - a. Satu macam ukuran bahan
 - b. Banyak ukuran bahan

2.3 Pola Pemotongan

1. *Guillotine Pattern*

Guillotine Pattern merupakan pola pemotongan yang dimulai dari satu sisi segi empat yang kemudian dilanjutkan

pada sisi lainnya. Pemotongan pertama dengan tipe *Guillotine Pattern* adalah dengan memotong bahan baku dengan panjang atau lebar yang sama. Pemotongan tersebut menghasilkan dua atau lebih *pieces* yang mempunyai panjang atau lebar yang sama, bukan kedua-duanya.

2. *Non-guillotine Pattern*

Pemotongan dengan tipe *non-guillotine* dilakukan apabila ukuran *pieces* yang diinginkan tidak memungkinkan untuk digabung dengan *pieces* yang lain.

3. Pola Dua Tahap Pemotongan (*Two Stage Pattern*)

Tahap pertama, pemotongan secara paralel atau pemotongan bahan secara horizontal, sehingga *rectangle* terbagi

2.4 Optimasi

2.4.1 Pembentukan Fungsi Tujuan

Sifat yang perlu diperhatikan dalam memilih kriteria untuk fungsi tujuan adalah sebagai berikut : [Ref. 3, hal. 80-81]

1. Lengkap
2. Operasional
3. Tidak Berlebihan
4. Minimum

2.4.2 Identifikasi Variabel

Dalam pemodelan, variabel yang teridentifikasi hendaknya dapat digolongkan menjadi empat jenis (Siegel dan Castellan, 1988) yaitu : [Ref. 3, hal. 94-95]

1. Variabel nominal
2. Variabel ordinal
3. Variabel interval
4. Variabel rasio

2.4.3 Uji Linearitas

Definisi dari fungsi linear adalah ”suatu fungsi $f(x_1, x_2, \dots, x_n)$ dari x_1, x_2, \dots, x_n adalah fungsi linear jika dan hanya jika untuk sejumlah set konstanta c_1, c_2, \dots, c_n berlaku fungsi $f(x_1, x_2, \dots, x_n) = c_1x_1 + c_2x_2 + \dots + c_nx_n$ ”. [Ref. 4, hal. 23]

2.5 *Integer Linear Programming*

Linear Programming merupakan metode atau teknik matematik yang digunakan untuk membantu dalam pengambilan keputusan. Di dalam *linear programming*, seluruh fungsinya (fungsi objektif serta fungsi pembatas) haruslah linear.

Terdapat empat asumsi dasar dalam penyelesaian masalah dengan menggunakan model *linear programming*, yaitu : [Ref. 5, hal. 38-44]

menjadi beberapa *rectangle* dengan panjang yang sama. Tahap kedua adalah pemotongan satu persatu bagian *rectangle*.

4. Pola Tiga Tahap Pemotongan (*Three Stage Pattern*)

Tahap pertama, pemotongan *rectangle* menjadi bagian-bagian dengan panjang atau lebar yang sama. Arah pemotongan tersebut dapat secara vertical maupun secara horizontal. Tahap kedua, hasil dari pemotongan tersebut dilanjutkan dengan pemotongan satu persatu yang terlebih dahulu mengubah arah pemotongan. Tahap ketiga, pemotongan dilakukan pada bagian yang menghasilkan *pieces*.

5. *One Group Guillotine Pattern*

1. *Proporsionalitas*.
2. *Divisibilitas*.
3. *Additivity*.
4. *Certainty*.

Integer Programming (IP) merupakan bentuk lain dari *Linear Programming* (LP) yang muncul karena tidak semua variabel keputusan dapat berupa bilangan pecahan dengan kata lain asumsi *divisibility* melemah atau hilang sama sekali.

Metode yang digunakan untuk memaksa pemecahan optimum dari *linear programming* yang dilonggarkan untuk bergerak ke arah pemecahan *integer* yang diinginkan adalah *branch & bound*. Algoritma *Branch & Bound* berlaku baik untuk masalah *integer* murni maupun masalah *integer* campuran. Keuntungan utamanya adalah bahwa batas atas tersebut dapat diestimasi dengan cepat dan dengan perhitungan minimal.

Tahapan yang dilakukan dalam algoritma *branch & bound* adalah sebagai berikut : [Ref. 4, hal. 217-227]

1. *Branching*
Apabila dari penyelesaian LP relaksasi diperoleh nilai variabel yang tidak *integer*, maka dilakukan *branching* atau pencabangan. Pencabangan dilakukan pada variabel yang bernilai pecahan atau tidak *integer*. Apabila terdapat lebih dari satu variabel yang bernilai pecahan, maka pilih secara sembarang (dari variabel pecahan tersebut) variabel yang akan dilakukan pencabangan.
2. *Bounding*
Setelah dilakukan *branching*, maka langkah selanjutnya adalah memilih salah satu subpersoalan yang belum diselesaikan dengan menerapkan aturan LIFO. Dari penghitungan yang dilakukan tersebut, diperoleh

nilai z untuk masing-masing subpersoalan. Nilai z ini dijadikan *bound*.

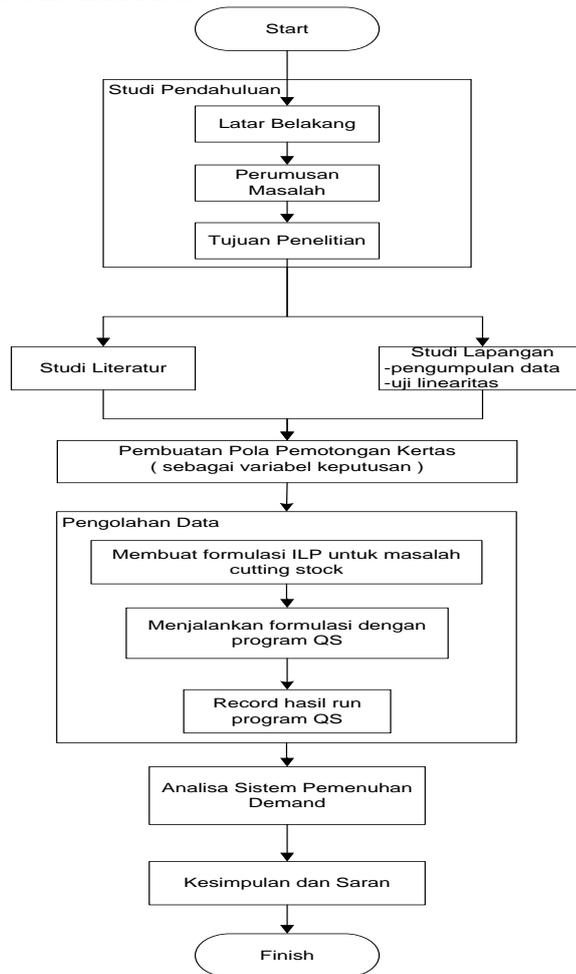
3. *Fathoming*
Ada tiga situasi yang menyebabkan suatu subpersoalan *fathomed*, yaitu :
 - a. Apabila subpersoalan tersebut tidak *feasible*.
 - b. Apabila subpersoalan itu memberikan solusi optimal dimana seluruh variabelnya berharga *integer*.
 - c. Apabila nilai z optimal untuk subpersoalan itu tidak lebih baik dari nilai z optimal subpersoalan lain (dalam persoalan maksimasi berarti nilai z optimal dari subpersoalan itu tidak lebih besar daripada batas bawah yang telah diperoleh).

Apabila subpersoalan berada dalam situasi *a* atau *c* maka subpersoalan tersebut dapat diabaikan atau dieliminasi dari pertimbangan selanjutnya.

2.6 Analisis Sensitivitas

Analisis sensitivitas adalah analisis yang dilakukan untuk mengetahui akibat/pengaruh dari perubahan yang terjadi pada parameter-parameter LP terhadap solusi optimal yang telah dicapai. Analisis sensitivitas sangat sesuai untuk mempelajari pengaruh variasi dalam koefisien fungsi objektif dan dalam jumlah sumber daya yang tersedia terhadap pemecahan optimum. [Ref. 4, hal. 106]

BAB III METODOLOGI PENELITIAN



Gambar 1 Metodologi penelitian

BAB IV PENGUMPULAN & PENGOLAHAN DATA

4.1 Pengumpulan Data

1. Data ukuran kertas *rectangle* (bahan baku).
2. Data ukuran kertas *pieces* serta banyaknya permintaan masing-masing *pieces*.
3. Data sisa kertas dari pola pemotongan saat ini.
4. Sistem pemenuhan permintaan saat ini.

4.2 Pengolahan Data

Langkah yang dilakukan adalah :

1. Menentukan ukuran dari persegi empat (bahan baku) serta *pieces*
2. Menghitung sisa kertas dari pola pemotongan awal.
3. Melakukan kombinasi pola pemotongan.
4. Menentukan sisa kertas dari kombinasi pola pemotongan.
5. Menentukan formulasi program linear, yaitu fungsi objektif dan fungsi pembatas.
6. Penyelesaian persoalan dengan program linear,
7. Menentukan *order fulfillment* untuk menerapkan pola pemotongan baru.

4.2.1 Formulasi Model

4.2.1.1 Fungsi tujuan

Minimasi sisa potong

$$\text{Min } Z = \sum_{j=1}^N \left\{ LW - \sum_{i=1}^m l_i w_i a_{ij} \right\} x_j$$

dimana :

L : panjang persegi empat (bahan baku)

W : lebar persegi empat (bahan baku)

l_i : panjang *piece i*

w_i : lebar *piece i*

a_{ij} : jumlah *piece i* pada pola j

Dengan nilai L, W, l_i dan w_i yang telah ditetapkan sebelumnya maka rumusan fungsi tujuan di atas dapat diringkas menjadi berikut :

$$\text{Min } Z = \sum_{j=1}^N S_j x_j$$

dimana, S_j : sisa potong dari pola j

4.2.1.2 Fungsi Pembatas

1. Fungsi Pembatas Batas Bawah Jumlah Pieces yang Dihasilkan

$$\sum_{j=1}^N a_{ij} x_j \geq d_{ib} \quad i = 1, 2, \dots, m \quad j = 1, 2, \dots, N$$

dimana, d_{ib} : batas bawah kebutuhan dari *piece i*

2. Pembatas Batas Atas Jumlah Pieces yang Dihasilkan

$$\sum_{j=1}^N a_{ij} x_j \leq d_{ia} \quad i = 1, 2, \dots, m \quad j = 1, 2, \dots, N$$

dimana, d_{ia} : batas atas kebutuhan dari *piece i*

3. Pembatas non-negatif

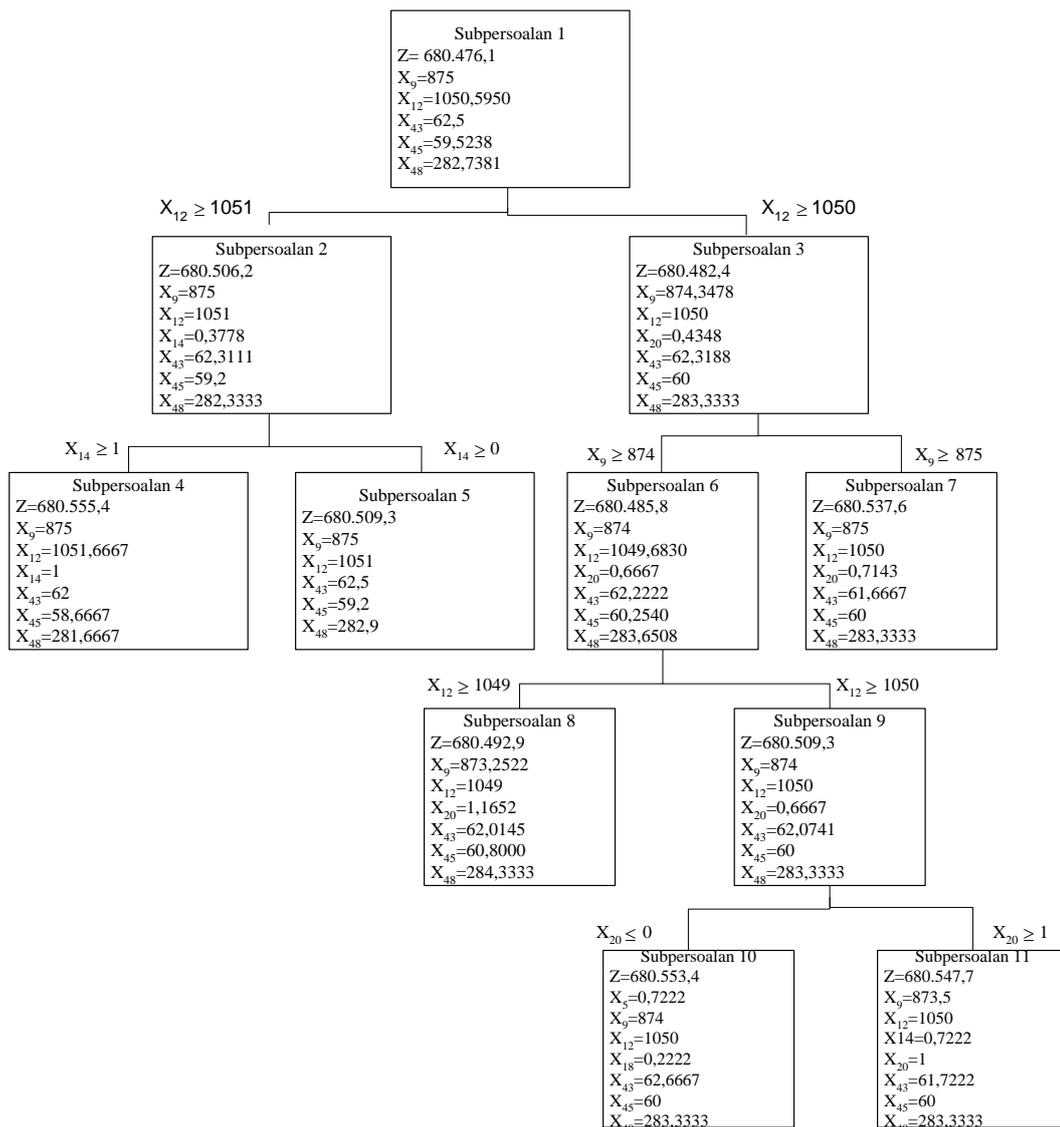
$$x_j \geq 0, \text{ integer}$$

$$j = 1, 2, \dots, N$$

4.2.1 Penyelesaian Formulasi dengan Algoritma *Branch & Bound* (manual)

Langkah pertama yang dilakukan adalah menyelesaikan formulasi di atas dengan mengabaikan pembatas *integer*,

dimana selanjutnya disebut dengan subpersoalan 1. Berikut adalah bagan gambaran langkah yang dilakukan dalam penyelesaian masalah dengan menggunakan Algoritma *Branch & Bound*.



Gambar 3 Bagan Penyelesaian Formula dengan Algoritma *Branch & Bound*

Demikian seterusnya hingga ditemukan solusi yang seluruh variabel keputusannya bernilai *integer* dengan nilai Z yang paling minimal. Pada bagan di atas telah dilakukan langkah *bounding* dan *branching* sedangkan *fathoming* belum dilakukan. Hal ini dikarenakan belum terdapat solusi yang seluruh variabelnya bernilai *integer*.

Apabila tidak terdapat permintaan terhadap salah satu atau dua tipe kertas yang ada (permintaan tipe-tipe kertas yang lain sesuai dengan rata-rata permintaan per hari) maka sebaiknya *Two Stage Guillotine Pattern* terpilih tidak diterapkan begitu saja. Hal ini dikarenakan adanya kemungkinan terjadinya penimbunan

terhadap tipe kertas tersebut. Begitupun juga dengan *One Group Guillotine Pattern* perbaikan, tidak dapat diterapkan dalam kondisi ini. Hal ini disebabkan oleh banyaknya sisa kertas yang terjadi.

BAB V PENUTUP

Dari penelitian yang telah dilakukan, dapat disimpulkan bahwa :

1. Komposisi pola pemotongan yang menghasilkan sisa kertas minimal dan dapat memenuhi kebutuhan kertas yang ada adalah hasil output dari algoritma yang disusun.
2. Produksi masing-masing alternatif pola pemotongan terpilih tersebut dapat

- diketahui pula dari output algoritma tersebut dengan menggunakan software QS LP-ILP.
3. Pada penelitian ini, diusulkan pula perubahan terhadap pola pemotongan *One Group Guillotine Pattern* yang digunakan saat ini. Pembuatan pola-pola tersebut masih tetap dengan metode *One Group Guillotine Pattern* hanya saja sisa kertasnya yang lebih minimal.
 4. Alternatif pola pemotongan *Two Stage Guillotine Pattern* terpilih dapat diterapkan apabila :
 - a) Apabila hanya dijumpai satu jenis tipe *pieces* yang dipesan, maka sebaiknya digunakan *One Group Guillotine Pattern* perbaikan.
 - b) Apabila *demand* yang datang sesuai dengan *demand* rata-rata per hari, maka sebaiknya digunakan pola-pola dari *Two Stage Guillotine Pattern* terpilih.
 - c) Apabila dijumpai pemesanan terhadap beberapa tipe *pieces* dengan tingkat *demand* kurang dari atau lebih dari *demand* rata-rata per hari, maka sebaiknya diterapkan metode kombinasi antara *Two Stage Guillotine Pattern* terpilih dengan *One Group Guillotine Pattern* perbaikan dalam pemenuhan *demand*nya.
 5. *Option* yang dapat dipilih oleh perusahaan :
 - a) Menggunakan *One Group Guillotine Pattern* perbaikan untuk seluruh permintaan yang datang.
 Positif : lead time tidak berubah
 Negatif : banyak sisa kertas terbuang
 - b) Menggunakan *One Group Guillotine Pattern* perbaikan, *Two Stage Guillotine Pattern* terpilih, dan pola kombinasinya sesuai dengan kondisi-kondisi permintaan yang telah dibahas di atas.
 Positif : sisa kertas terbuang lebih sedikit
 Negatif : lead time lebih lama

Berdasarkan analisis sensitivitas, *Two Stage Guillotine Pattern* terpilih hanya optimal pada tingkat

permintaan tertentu saja. Di luar jumlah tersebut, *Two Stage Guillotine Pattern* terpilih bukanlah solusi yang optimal lagi. Diperlukan bantuan *software* untuk dapat mengetahui solusi yang optimalnya. Disarankan untuk tetap menggunakan *Two Stage Guillotine Pattern* terpilih dan kekurangannya ditutup dengan menggunakan *One Group Guillotine Pattern* perbaikan.

- c) Mengubah sistem produksi dari *Make to Order* menjadi *Make to Stock* dengan menggunakan *Two Stage Guillotine Pattern* terpilih.
 Positif : sisa kertas terbuang lebih sedikit dan lead time tidak lama
 Negatif : menyediakan ruangan penyimpanan

REFERENSI

- [1] Scheithauer, G and G. Belov, A *Branch-and-Cut-and-Price Algorithm for One-Dimensional Stock Cutting and Two-Dimensional Two-Stage Cutting*, Technical Report MATH-NM-03, Dresden University, 2003
- [2] Cung, Van-Dat, Mhand Hifi and Bertrand Le Cun, *Constrained Two-Dimensional Cutting Stock Problems a Best-First Branch and Bound Algorithm*, International Transactions In Operational Research, No. 7, pp.185-210. 2000
- [3] Simatupang, Togar M., *Pemodelan Sistem*, Nindita : Klaten, 1995
- [4] Dimiyati, Tjutju Tarliah dan Ahmad Dimiyati, *Operations Research : Model-model Pengambilan Keputusan*, Sinar Baru Algensindo : Bandung, 2003
- [5] Lieberman, Gerald J. and Frederick S, Hillier, *Introduction to Operations Research-6th ed*, McGraw Hill : New York, 1995
- [6] Nonas, Sigrid Lise and Anders Thorstenson, *A Combined Cutting Stock and Lot Sizing Problem*, European Journal of Operational

- Research, No. 120, pp.327-342. 2000
- [7] Taha, Hamdy A., *Operations Research:An Introduction-3rd ed*, Macmillan Publishing Co., Inc. : New York, 1982
- [8] Teo,Chung-Piaw and Jay Sethuraman, *Theory and Methodology On Cutting Plane Heuristic for The Stabel Roommates Problem and Its Applications*, European Journal of Operational Research, No.123, pp.195-205. 2000
- [9] Chang, Pei-Chann and Jih-Chang Hsieh, *An Investigation of Paper Cutting Problem by Dynamic Programming and Heuristic Approaches*, Journal of the Chinese Institute of Industrial Engineers, Vol. 22, No. 6, pp. 463-472. 2005