

---

## Penyelesaian Masalah Pemotongan Rol Kertas dengan Metode Penghasil Kolom

Rosa Ajeng Mahadika<sup>1</sup>, Hartono<sup>2</sup>

<sup>1,2</sup>Program Studi Matematika, Fakultas Sains dan Teknologi, Universitas Sanata Dharma  
Kampus III Paingan, Maguwoharjo, Depok, Sleman, Yogyakarta  
Telp. (0274) 513301, 515352 Fax. (0274) 562383  
e-mail: 123114003@student.usd.ac.id, ajengmahadika@gmail.com

### Abstract

*Paper industry produces paper rolls called jumbo rolls using paper machines. The paper rolls are then cut into rolls with different widths. The widths of rolls are determined by the customers' demands and the different number of ordered rolls so that it is necessary to have an organization of cutting pattern from a jumbo into small rolls. The organization of cutting pattern aims to minimize the number of jumbo rolls used and rest of cutting.*

*This research implements a column generation method to solve the problem. The column generation method is one of the linear programming techniques for the problem of supplies cutting. The iteration of column generation method uses revised simplex and knapsack problem with the completion of branch-and-bound method. If a solution is not an integer, the solution is converted into the integer using the first-fit decreasing method. Then, a display program with MATLAB is made based on the column generation algorithm. In this program, the solution may be in form of the number of rolls or the weight of rolls.*

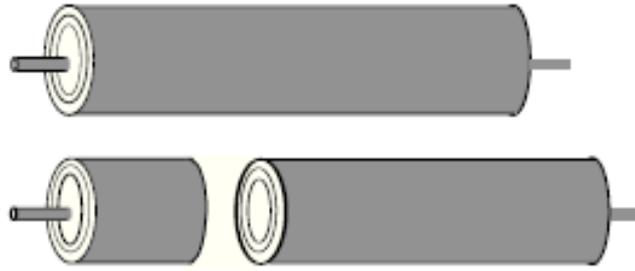
*Numerical examples are then carried out to show the effectiveness of the method. Based on the result of the simulation, the optimal solution is obtained, namely the minimum number of jumbo rolls and rest for several incoming order cases. In comparison to the manual calculation commonly practiced by paper industry, the results are well fitted. However, for big problems our approach is better because manual calculation is almost impossible to be done due to the expanding number of possible cutting pattern combinations.*

**Keywords:** paper roll, cutting pattern, linear programming, knapsack problem, column generation

### 1. Pendahuluan

Masalah pemotongan persediaan (*cutting stock*) sering terjadi pada proses produksi. Masalah pemotongan persediaan biasanya berkaitan dengan pemakaian bahan baku yang optimal yaitu yang meminimumkan biaya produksi bahan baku. Pada industri kertas, untuk dapat meminimumkan biaya produksi salah satu cara yang ditempuh adalah dengan memproduksi jumlah rol yang optimal dalam arti yang sesuai dengan kebutuhan/pesanan. Semakin sedikit jumlah rol dan semakin baik pola pemotongan yang dihasilkan maka efisiensi akan meningkat.

Sebelum rol jumbo dipotong menjadi potongan kertas atau rol kecil maka harus diperhitungkan berbagai macam kemungkinan pola pemotongan dari rol jumbo tersebut yang kemudian akan dipilih yang paling optimal. Pola tersebut berupa gabungan dari beberapa ukuran kertas atau rol yang diinginkan nasabah. Pembentukan pola tersebut juga harus memperhatikan beberapa kendala agar didapat hasil yang optimal. Beberapa kendala yang harus diperhatikan misalnya adalah lebar kertas maksimal (*deckle*) 276 cm untuk 70 gsm (*grams per square meter*) ke atas dan maksimal 272 untuk 70 gsm ke bawah, serta dalam 1 rol jumbo yang akan dipotong menjadi rol-rol kecil haruslah memuat pesanan dengan panjang rol, jenis kertas, warna kertas, dan gsm yang sama.



**Gambar 1.** Pemotongan rol jumbo menjadi beberapa bagian (rol kecil).

Terdapat banyak cara untuk menyelesaikan masalah tersebut [4], salah satunya program linear dengan metode simpleks. Secara umum, masalah program linear dapat dirumuskan sebagai berikut:

Maksimumkan atau minimumkan

$$z = \mathbf{c}\mathbf{x} \quad (1)$$

Dengan kendala

$$\begin{aligned} \mathbf{A}\mathbf{x} &= \mathbf{b} \\ \mathbf{x} &\geq \mathbf{0} \end{aligned} \quad (2)$$

dengan  $\mathbf{x} = (x_1, x_2, \dots, x_{n+m})^T$ ,  $\mathbf{c} = (c_1, c_2, \dots, c_{n+m})$ ,  $\mathbf{A} = \begin{bmatrix} a_{11} & \dots & a_{1,n+m} \\ \vdots & \ddots & \vdots \\ a_{m1} & \dots & a_{m,n+m} \end{bmatrix}$ , dan

$\mathbf{b} = \begin{bmatrix} b_1 \\ \vdots \\ b_m \end{bmatrix}$ . Untuk lebih detilnya dapat dilihat [1].

Misalkan terdapat  $n$  kemungkinan pola pemotongan untuk rol jumbo dengan lebar  $r$ , rol kecil memiliki lebar  $w_i$  untuk  $i = 1, \dots, m$ , dan  $p_i$  adalah banyaknya rol kecil dengan lebar  $w_i$  ( $p_i$  bilangan bulat non negatif) sehingga  $\sum_{i=1}^m p_i w_i \leq r$ . Maka masalah pemotongan ini dapat diselesaikan dalam program linear sebagai berikut.

Minimumkan

$$z = \sum_{j=1}^n x_j \quad (3)$$

Dengan kendala

$$\begin{aligned} \sum_{j=1}^n p_{ij} x_j &\geq b_i \\ x_j &\geq 0 \end{aligned} \quad (4)$$

dan  $p_{ij}$  adalah banyaknya rol kecil dengan lebar  $w_i$  dalam pola pemotongan ke- $j$ ,  $b_i$  adalah banyaknya permintaan rol kecil dengan lebar  $w_i$ , variabel  $x_j$  menunjukkan banyaknya rol jumbo yang dipotong pada pemotongan ke- $j$ .

Sebagai contoh, akan diberikan permasalahan yang diambil dari [8].

#### Contoh 1.1

Sebuah industri kertas menghasilkan rol jumbo dengan lebar 91 inchi dan pelanggan memesan sejumlah rol dengan lebar yang lebih kecil seperti tampak dalam tabel 1.

Banyak rol	Lebar rol (inchi)
78	25.5
40	22.5
30	20
30	15

Permasalahannya menjadi bagaimana menentukan pola pemotongan rol jumbo agar pesanan dapat dipenuhi dengan banyaknya rol jumbo yang harus dipotong sesedikit mungkin.

**Tabel 2.** Kemungkinan Pola Pemotongan

Pola	Lebar rol				Sisa
	25.5	22.5	20	15	
1	3	0	0	0	14.5
2	2	1	0	1	2.5
3	2	0	2	0	0
4	2	0	0	2	10
5	2	0	1	1	5
6	1	2	1	0	0.5
7	1	2	0	1	5.5
8	1	1	2	0	3
9	1	1	1	1	8
10	1	1	0	2	13
11	1	0	3	0	5.5
12	1	0	2	1	10.5
13	1	0	1	3	0.5
14	1	0	0	4	5.5
15	0	4	0	0	1
16	0	3	1	0	3.5
17	0	3	0	1	8.5
18	0	2	2	0	6
19	0	2	1	1	11
20	0	2	0	3	1
21	0	1	3	0	8.5
22	0	1	2	1	13.5
23	0	1	1	3	3.5
24	0	1	0	4	8.5
25	0	0	4	0	11
26	0	0	3	2	1
27	0	0	2	3	6
28	0	0	1	4	11
29	0	0	0	6	1

Kemungkinan/cara memotong rol jumbo ke dalam rol kecil sesuai pesanan (dengan sisa pemotongan kurang dari 15 inchi) dapat dilihat dalam tabel 2.

Pola 1 dari tabel di atas berarti 1 rol jumbo dengan lebar 91 inchi akan dipotong menjadi 3 rol kecil dengan lebar 25.5 inchi sehingga tersisa 14.5 inchi. Pola 2 berarti 1 rol jumbo akan dipotong menjadi 2 rol kecil dengan lebar 25.5, 1 rol kecil dengan lebar 22.5 dan 1 rol kecil dengan lebar 15 inchi sehingga menghasilkan sisa pemotongan 2.5 inchi. Demikian seterusnya berlaku cara membaca data yang sama untuk pola-pola pemotongan yang lain.

Untuk setiap kemungkinan pola  $P_j$  di atas, kita memperkenalkan variabel  $x_j \geq 0$  yang menunjukkan banyaknya rol jumbo yang harus dipotong menurut pola  $P_j$ . Dengan demikian,

fungsi tujuan adalah meminimumkan jumlah rol jumbo yang dipotong yaitu  $\sum_{j=1}^{29} x_j$ . Agar pesanan terpenuhi maka untuk setiap ukuran lebar yang dipesan ditambahkan 1 kendala. Sebagai contoh, untuk pesanan 78 rol dengan lebar 25,5 inchi, maka kendala dapat dituliskan

$$3x_1 + 2x_2 + 2x_3 + 2x_4 + 2x_5 + x_6 + x_7 + x_8 + x_9 + x_{10} + x_{11} + x_{12} + x_{13} + x_{14} \geq 78$$

yang berarti jumlah rol kecil dengan lebar 25.5 inchi yang dihasilkan dengan memotong rol jumbo menurut berbagai pola pemotongan tidak boleh kurang dari 78 rol (jumlah rol pesanan). Demikian seterusnya sehingga diperoleh masalah program linear berikut:

Minimumkan

$$\sum_{j=1}^{29} x_j \quad (5)$$

Dengan kendala

$$\begin{aligned} 3x_1 + 2x_2 + 2x_3 + 2x_4 + 2x_5 + x_6 + x_7 + x_8 + x_9 + x_{10} + x_{11} + x_{12} + x_{13} + x_{14} &\geq 78 \\ x_2 + 2x_6 + 2x_7 + x_8 + x_9 + x_{10} + 4x_{15} + 3x_{16} + 3x_{17} + 2x_{18} + 2x_{19} + 2x_{20} + x_{21} + x_{22} \\ &+ x_{23} \\ &+ x_{24} \geq 40 \\ 2x_3 + x_5 + x_6 + 2x_8 + x_9 + 3x_{11} + 2x_{12} + x_{13} + x_{16} + 2x_{18} + x_{19} + 3x_{21} + 2x_{22} + x_{23} \\ &+ 4x_{25} \\ &+ 3x_{26} + 2x_{27} + x_{28} \geq 30 \\ x_2 + 2x_4 + x_5 + x_7 + x_9 + 2x_{10} + x_{12} + 3x_{13} + 4x_{14} + x_{17} + x_{19} + 3x_{20} + x_{22} + 3x_{23} \\ &+ 4x_{24} \\ &+ 2x_{26} + 3x_{27} + 4x_{28} + 6x_{29} \geq 30 \\ x_j &\geq 0 \end{aligned}$$

Masalah tersebut dapat diselesaikan dengan program *QM for Windows* yang merupakan perangkat lunak digunakan untuk membantu proses perhitungan secara teknis pengambilan keputusan secara kuantitatif. Program ini menyediakan modul-modul dalam area pengambilan keputusan bisnis seperti *assignment*, *forecasting*, *integer programming*, *linear programming*, *quality control*, *inventory*, dan lain-lain. Lihat [2] untuk penjelasan yang lebih detail.

Gambar 2 adalah hasil yang didapat menggunakan QM.

Variable	Status	Value
X1	NONBasic	0
X2	Basic	24
X3	Basic	15
X4	NONBasic	0
X5	NONBasic	0
X6	NONBasic	0
X7	NONBasic	0
X8	NONBasic	0
X9	NONBasic	0
X10	NONBasic	0
X11	NONBasic	0
X12	NONBasic	0
X13	NONBasic	0
X14	NONBasic	0
X15	Basic	4
X16	NONBasic	0
X17	NONBasic	0
X18	NONBasic	0
X19	NONBasic	0
X20	NONBasic	0
X21	NONBasic	0
X22	NONBasic	0
X23	NONBasic	0
X24	NONBasic	0
X25	NONBasic	0
X26	NONBasic	0
X27	NONBasic	0
X28	NONBasic	0
X29	Basic	1
surplus 1	NONBasic	0
surplus 2	NONBasic	0
surplus 3	NONBasic	0
surplus 4	NONBasic	0
Optimal Value (Z)		44

**Gambar 2.** Tampilan program konversi rol

Dari gambar di atas, didapatkan solusi optimal yaitu  $x_2 = 24$ ,  $x_3 = 15$ ,  $x_{15} = 4$ ,  $x_{29} = 1$  dan selainnya bernilai 0. Itu berarti untuk memenuhi pesanan diperlukan sebanyak 24 rol untuk pola pemotongan kedua, 15 rol untuk pola pemotongan ketiga, 4 rol untuk pola pemotongan ke lima belas dan 1 rol untuk pola pemotongan ke dua puluh sembilan. Dengan demikian, banyaknya rol jumbo yang digunakan sebanyak 44 rol.

Namun pada masalah nyata di industri kertas, banyaknya dan jenis pesanan akan sangat beragam sehingga masalah ini tidak mungkin diselesaikan secara manual (menyusun tabel kemungkinan pemotongan kemudian diselesaikan dengan program linear). Masalah lain yang mungkin muncul dan tidak mudah diselesaikan adalah solusi yang didapatkan belum tentu merupakan bilangan bulat sehingga diperlukan cara tertentu untuk mengubah solusi tersebut menjadi bilangan bulat. Dalam kasus rol kecil yang dipesan jumlahnya tidak banyak, maka pola yang digunakan pada solusi optimal bilangan bulat mungkin berbeda dengan solusi optimal aslinya (dalam pecahan). Oleh karena itu, dalam artikel ini dibahas metode penghasil kolom (*column generation*) yang dapat menyelesaikan masalah pemotongan secara lebih efisien.

## 2. Penghasil Kolom

Metode penghasil kolom adalah suatu metode untuk menemukan himpunan dari pola pemotongan optimum pada masalah pemotongan persediaan. Lihat [5], [6] dan [7] untuk uraian lebih mendalam dan aplikasi metode penghasil kolom. Dalam metode ini, pada dasarnya, setiap pola merupakan suatu kolom dari masalah program linearnya. Pada masalah nyata, banyaknya pola pemotongan dapat menjadi sangat banyak. Daripada mempertimbangkan banyaknya kemungkinan pola pemotongan, metode penghasil kolom bekerja dengan membangun suatu model bagian dari masalah pemotongan persediaan yang secara sistematis menghasilkan pola baru sehingga solusi optimum dapat dicapai. Pola baru ini ditambahkan ke model bagian dengan program bantuan bilangan bulat [3].

Model bagian pemotongan persediaan dapat dimulai dengan banyak cara. Pilihan termudah adalah memasukkan satu pola untuk setiap ukuran rol. Setiap pola terdiri dari maksimum banyaknya rol yang dapat dipotong dari rol jumbo.

Diasumsikan terdapat beberapa pola pemotongan  $\mathbf{a}_j$  yang bukan bagian dari model bagian pemotongan persediaan. Misalkan  $a_i$  komponen dari vektor  $\mathbf{a}_j$ . Setiap komponen berkorespondensi dengan banyaknya rol ukuran  $i$  yang digunakan pada pola pemotongan. Misalkan  $y_i$  adalah koefisien pada fungsi tujuan yang berhubungan dengan setiap keperluan permintaan  $i$  pada model bagian pemotongan persediaan. Maka pola pemotongan  $\mathbf{a}_j$  yang harus ditambahkan ke model bagian sewaktu-waktu adalah

$$1 - \sum_{i=1}^r y_i a_i < 0 \quad (6)$$

Kondisi ini adalah syarat optimal pada metode simpleks direvisi ketika diaplikasikan ke model pola pemotongan persediaan.

Perhatikan masalah pemotongan persediaan dimana rol jumbo berukuran  $r$  dan banyaknya pesanan tiap rol kecil  $b_i$  dengan lebar  $w_i$  ( $i = 1, 2, \dots, m$ ). Masalah program linear dapat dituliskan seperti ini.

Minimumkan  $\mathbf{z} = \mathbf{c}\mathbf{x}$

Dengan kendala  $\mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq 0$

Dimana  $\mathbf{b}$  adalah vektor kolom dengan komponen  $b_1, b_2, \dots, b_m$  dan  $\mathbf{c}$  adalah vektor baris dengan komponen  $1, 1, \dots, 1$ . Setiap kolom  $\mathbf{a}_j = [a_{1j}, a_{2j}, \dots, a_{mj}]^T$  dari  $\mathbf{A}$  menunjukkan pola pemotongan rol jumbo menjadi rol kecil  $a_i$  dengan lebar  $w_i$  ( $i = 1, 2, \dots, m$ ). Jadi  $\mathbf{a}_j$  adalah kolom dari  $\mathbf{A}$  jika hanya jika  $a_{1j}, a_{2j}, \dots, a_{mj}$  adalah bilangan bulat non negatif sedemikian hingga  $\sum w_i a_i \leq r$ . Dengan metode simpleks yang direvisi ditunjukkan adanya kolom non dasar dari  $\mathbf{A}$  di langkah 2 dari setiap iterasi, yaitu ketika kolom baru  $\mathbf{a}_j$  (kolom masuk) ditemukan. Setelah menghitung vektor baris  $\mathbf{y}$  (harus berupa bilangan bulat nonnegatif), kita mencari  $a_1, a_2, \dots, a_m$  bilangan bulat non negatif sedemikian hingga

$y_i \geq 0$ , untuk setiap bilangan bulat  $i$

$$\sum_{i=1}^r w_i a_i \leq r \quad (7)$$

$$1 - \sum_{i=1}^r y_i a_i < 0 \quad (8)$$

Ketika  $\sum_{i=1}^r y_i a_i > 1$ , maka pertidaksamaan terpenuhi. Pertidaksamaan ini dapat ditulis sebagai fungsi tujuan untuk memaksimalkan kendala pertama dari formulasi model di atas. Ketika nilai optimal dari program matematika lebih besar dari satu, maka pola pemotongan ditemukan. Ketika nilai optimal kurang dari atau sama dengan satu, maka tidak terdapat pola pemotongan yang dapat meningkatkan nilai tujuan dari masalah pemotongan persediaan. Sehingga model penghasil pola pemotongan dapat dituliskan sebagai berikut.

Maksimumkan  $z = \sum_{i=1}^m y_i a_i$

Dengan kendala

$$\sum_{i=1}^m w_i a_i \leq r \quad (9)$$

$y_i \geq 0$ , untuk setiap bilangan bulat  $i$

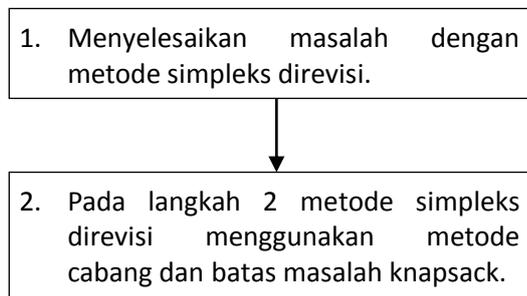
Model ini yang nantinya akan diselesaikan menggunakan masalah knapsack.

Langkah-langkah dari penghasil kolom merupakan gabungan dari langkah metode simpleks direvisi dengan knapsack yaitu:

1. Menyelesaikan masalah dengan metode simpleks direvisi.

2. Pada langkah kedua di setiap iterasi metode simpleks direvisi dihitung dengan metode cabang dan batas masalah knapsack.

Diagram alir untuk masalah metode penghasil kolom dapat dilihat pada gambar 3.



**Gambar 3.** Diagram alir penghasil kolom

### 3. Metode Simpleks Direvisi

Pendekatan umum dari metode simpleks direvisi adalah memperoleh suatu urutan solusi-solusi layak dasar yang semakin baik sampai tercapai suatu solusi optimal. Salah satu ciri pokok dari metode simpleks direvisi mencakup dengan cara mana setiap solusi layak dasar akan diselesaikan, yaitu setelah variabel-variabel dasar dan non dasar diketahui.

Untuk setiap solusi layak dasar  $x^*$  yaitu  $x_1, x_2, \dots, x_{n+m}$  dibagi ke dalam variabel dasar  $m$  dan variabel non dasar  $n$ . Contohnya, membagi matriks  $A$  menjadi  $A_B$  dan  $A_N$ , membagi vektor  $x$  menjadi  $x_B$  dan  $x_N$ , dan membagi  $c$  menjadi  $c_B$  dan  $c_N$ . Sehingga kita dapat menuliskan  $Ax = b$  menjadi

$$A_B x_B = b - A_N x_N \quad (10)$$

Dimana matriks  $A_B$  adalah nonsingular.

Matriks  $A_B$  disebut juga matriks basis atau basis. Matriks basis  $A_B$  dapat kita notasikan menjadi matriks  $B$ . Sehingga kita dapat menuliskan persamaan  $x_B$  dan  $z$  menjadi  $x_B = B^{-1}b - B^{-1}A_N x_N$  dan  $z = c_B B^{-1}b + (c_N - c_B B^{-1}A_N)x_N$ . Tentunya  $B^{-1}b$  tak lain vektor  $x_B^*$  yang menentukan nilai sekarang dari variabel dasar.

Berikut iterasi dari metode simpleks direvisi yaitu sebagai berikut.

1. Selesaikan sistem  $yB = c_B$  dimana  $B$  adalah matriks basis awal, sehingga ditemukan vektor  $y$ .
2. Tentukan kolom yang masuk, yaitu jika variabel tidak dasar  $x_j$  berhubungan dengan elemen  $c_j$  dari  $c_N$  dan kolom  $a$  dari  $A_N$ , maka  $c_N - yA_N = c_j - ya$ . Untuk masalah maksimum (minimum), kolom  $a$  dipilih yang memiliki  $c_j - ya$  paling positif (negatif). Untuk masalah maksimum (minimum) jika semua elemen  $c_j - ya < 0$  ( $c_j - ya > 0$ ), maka tidak terdapat kolom masuk dan iterasi berhenti sehingga didapatkan solusi optimal. Jika tidak, maka lanjut ke langkah 3.
3. Selesaikan sistem  $Bd = a$ , sehingga didapat vektor  $d$ .
4. Tentukan kenaikan nilai  $t$  terbesar dari nol sampai suatu nilai positif dengan cara mencari nilai paling minimum dari  $\frac{x_B^*}{d}$  sedemikian hingga  $x_B^* - td \geq 0$ . Jika tidak terdapat nilai  $t$  atau terdapat elemen di  $d \leq 0$ , maka solusi optimal tak terbatas atau tidak memiliki penyelesaian. Jika terdapat elemen yang memenuhi  $x_B^* - td = 0$  maka kolom tersebut menjadi kolom keluar.
5. Menukar kolom keluar dari  $B$  dengan kolom masuk dan tukar variabel keluar dengan variabel masuk. Lalu kembali pada langkah 1 sampai solusi optimal diperoleh.

Diagram alir metode simpleks direvisi dapat dilihat pada gambar 4.

#### 4. Masalah Knapsack

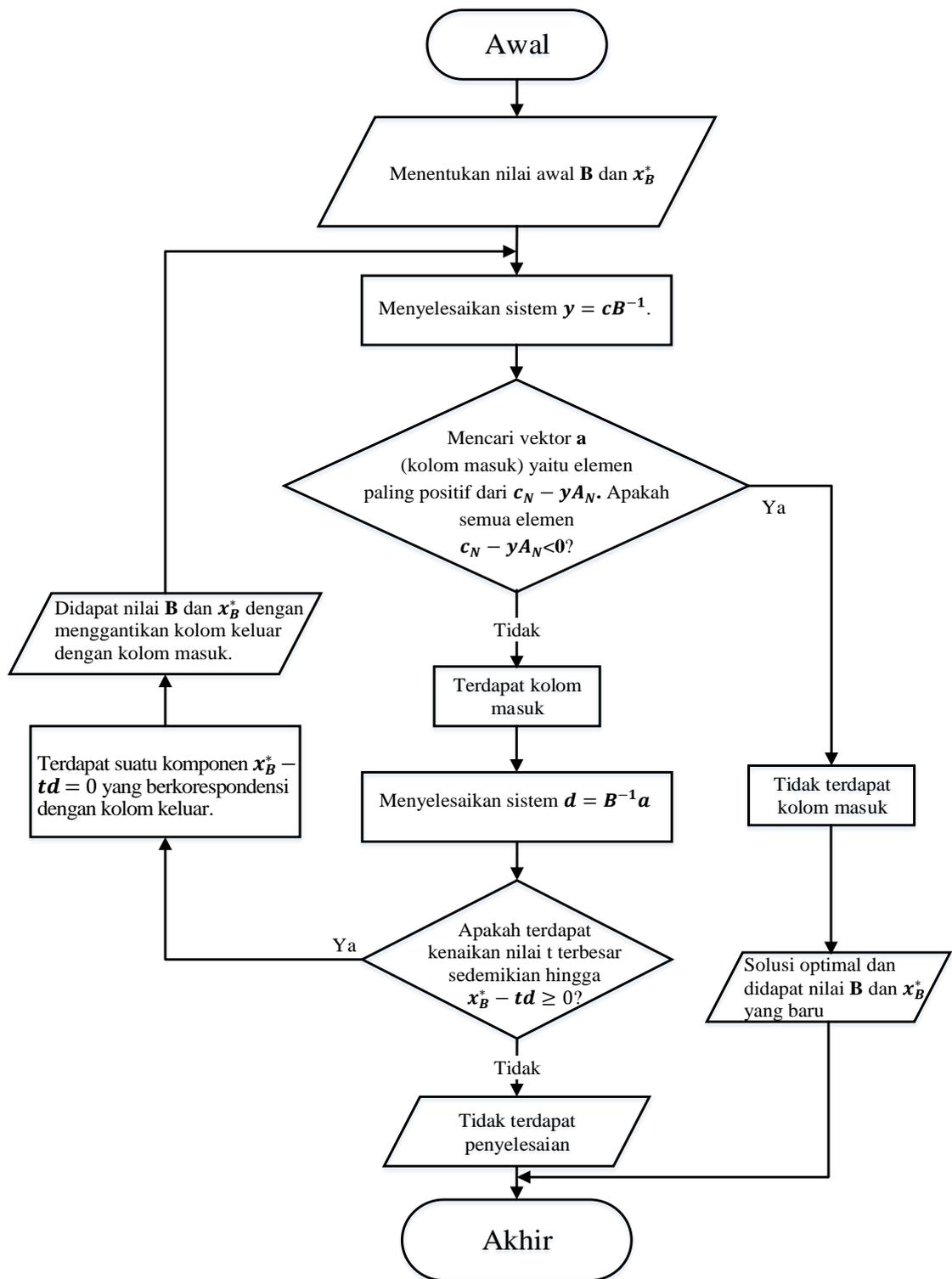
Seperti yang sudah dibahas pada subbab sebelumnya bahwa dalam menentukan pola pemotongan diperlukan masalah knapsack sebagai penyelesaiannya [8]. Masalah knapsack dalam masalah pemotongan persediaan (rol kertas) dapat dituliskan sebagai berikut.

Maksimumkan  $z = \sum_{i=1}^m y_i a_i$

Dengan kendala

$$\sum_{i=1}^m w_i a_i \leq r \quad (11)$$

Dimana  $a_i$  adalah bilangan bulat non negatif ( $i = 1, 2, \dots, m$ ) dan merupakan banyaknya rol ke- $i$  yang dapat dibuat pada rol,  $w_i$  adalah ukuran rol ke- $i$  yang bernilai positif dan  $r$  adalah ukuran rol jumbo yang bernilai positif.



Gambar 4. Diagram alir metode simpleks direvisi

Berikut adalah tahapan untuk menyelesaikan masalah knapsack dengan metode cabang dan batas.

1. Menentukan nilai awal yaitu  $M = 0$  dan  $k = 0$ .
2. Menemukan perpanjangan cabang. Untuk  $j = k + 1, k + 2, \dots, m$  maka  $a_j = \lfloor (r - \sum_{i=1}^{j-1} w_i a_i) / w_j \rfloor$ . Biasanya untuk  $a_1 = \lfloor r / w_1 \rfloor$ . Maka didapat solusi terbaik  $a_1^*, a_2^*, \dots, a_m^*$ .
3. Memperbaiki solusi. Jika  $\sum_{i=1}^m y_i a_i > M$ , maka mengganti  $M$  dengan  $\sum_{i=1}^m y_i a_i$  dan mengganti  $a_1^*, a_2^*, \dots, a_m^*$  dengan  $a_1, a_2, \dots, a_m$ .
4. Menemukan cabang selanjutnya.  
Menemukan  $k$  terbesar sedemikian hingga  $k \leq m - 1$  dimana  $a_k > 0$ . Kita dapat tuliskan  $\bar{a}_i = a_i$  untuk  $i = 1, 2, \dots, k - 1$ .
  - a. Jika  $k = 1$  maka berhenti; selain itu ganti  $k$  dengan  $k - 1$ .
  - b. Jika  $a_k = 0$ , maka kembali ke 4a, selain itu ganti  $a_k$  dengan  $a_k - 1$ .
5. Pencarian cabang yang lebih baik.

Jika

$$\sum_{i=1}^k y_i \bar{a}_i + \frac{y_{k+1}}{w_{k+1}} (r - \sum_{i=1}^k w_i \bar{a}_i) \leq M \quad (\text{untuk koefisien } y_i \text{ bukan bilangan bulat positif})$$

atau

$$\sum_{i=1}^k y_i \bar{a}_i + \frac{y_{k+1}}{w_{k+1}} (r - \sum_{i=1}^k w_i \bar{a}_i) \leq M + 1 \quad (\text{untuk koefisien } y_i \text{ bilangan bulat positif})$$

maka  $\bar{a}_1, \bar{a}_2, \dots, \bar{a}_k$  tidak layak diperiksa. Oleh karena itu, harus kembali ke langkah 4. Selain itu, kembali ke langkah 2.

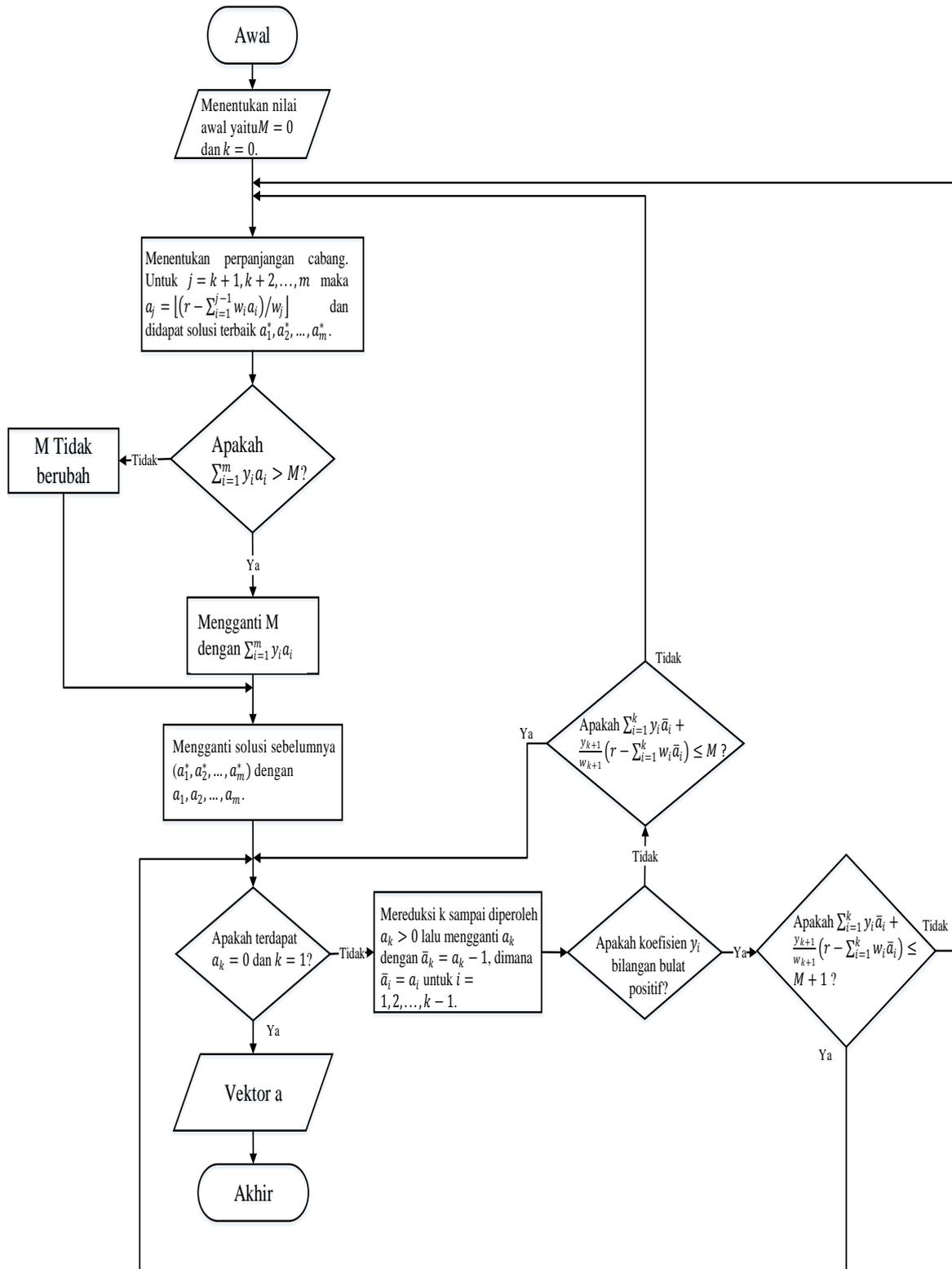
Diagram alir untuk penyelesaian masalah Knapsack dapat dilihat pada gambar 5.

Jika, solusi optimal yang dihasilkan bukanlah bilangan bulat, maka diperlukan suatu metode untuk memberikan solusi berupa bilangan bulat [8]. Metode yang digunakan adalah *first-fit decreasing*. Pada itersi ke- $j$  dari metode ini yaitu menemukan pola pemotongan rol jumbo ke- $j$ . ITERS dimulai dengan sisa permintaan setelah jumlah rol dibulatkan ke bawah yaitu  $b'_1, b'_2, \dots, b'_m$ . Pola pemotongan yang dihasilkan untuk setiap iterasi yaitu

$$a_i = \min \left\{ \begin{array}{l} b'_i \\ \lfloor (r - \sum_{k=1}^{i-1} w_k a_k) / w_i \rfloor \end{array} \right. \quad (12)$$

Untuk  $i = 1, 2, \dots, m$ , kemudian ganti setiap nilai  $b'_i$  dengan  $b'_i - a_i$  dan lanjutkan proses iterasi ke- $j+1$ .

Pada penelitian ini akan dibahas mengenai bagaimana penerapan penghasil kolom dalam menyelesaikan masalah pemotongan rol kertas untuk mendapatkan solusi optimal. Masalah pemotongan rol kertas dibatasi hanya pada pemotongan dari rol ke rol yang berarti hanya untuk pemotongan dari rol jumbo menjadi rol-rol kecil dan dengan pola pemotongan satu dimensi. Yang dimaksud pola pemotongan satu dimensi yaitu dengan mempertimbangkan satu ukuran saja (lebar) sedangkan untuk tebal/diameter dan panjang rol adalah sama untuk setiap pesanan.



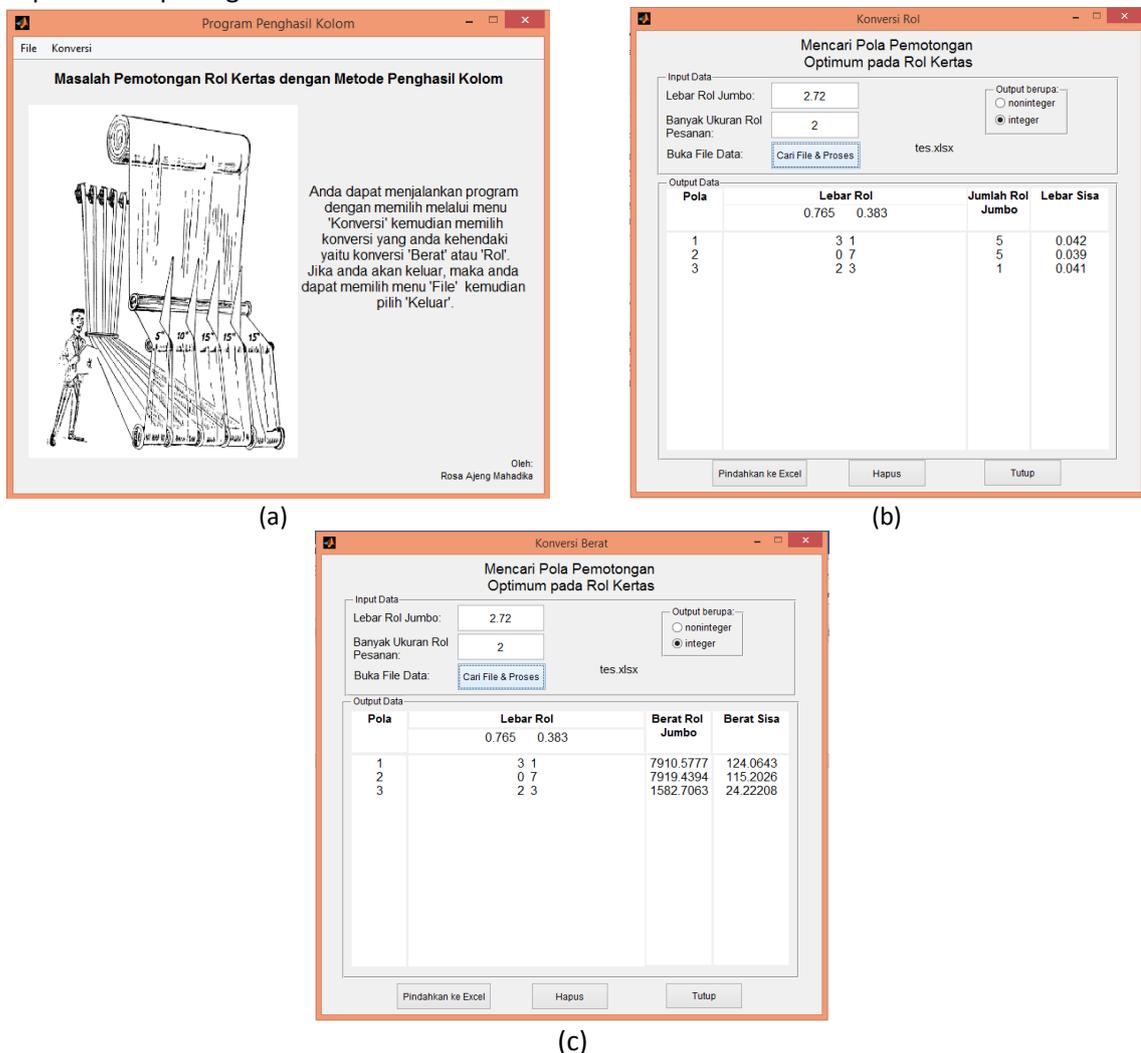
Gambar 5. Diagram alir masalah knapsack

## 5. Metode Penelitian

Penelitian ini dilakukan yaitu studi pustaka, yaitu dengan mempelajari buku atau jurnal yang berkaitan dengan masalah pemotongan persediaan (*cutting stock problem*). Penulis juga menggunakan studi kasus untuk memperoleh data yang akan digunakan dalam penelitian. Pengumpulan data dilakukan dengan wawancara pihak perusahaan dan melihat data dari perusahaan berupa data permintaan produksi rol kertas, data kertas (ukuran, jenis, warna, dan gsm), dan data historis produksi bulan Juli sampai September 2016. Karena penelitian ini hanya dibatasi pada pemotongan dari rol ke rol dan dengan pola pemotongan satu dimensi maka data yang diperoleh harus dikelompokkan berdasarkan ukuran jenis, warna, dan gsm kertas.

## 6. Hasil dan Pembahasan

Untuk mempermudah proses penyelesaian menggunakan metode penghasil kolom maka dibuat suatu program tampilan MATLAB. Program yang dibuat menampilkan hasil yang dapat berupa konversi rol dan konversi berat. Tampilan program awal, konversi rol dan berat dapat dilihat pada gambar 6.



**Gambar 6.** Tampilan program: (a) Tampilan awal program, (b) konversi rol, (c) konversi berat

Kasus 1.

Pada contoh sebelumnya telah diselesaikan dengan memodelkan masalah program linear lalu diproses pada QM, pada kasus ini diselesaikan contoh tersebut dengan menggunakan penghasil kolom. Sehingga didapatkan hasil seperti berikut.

**Tabel 3.** Hasil dari Kasus 1 dengan Metode Penghasil Kolom

Pola pemotongan ke-	Lebar rol				Banyak rol
	25,5	22,5	20	15	
1	2	1	0	1	24
2	0	4	0	0	4
3	2	0	2	0	15
4	0	0	0	6	1

Untuk kasus berikutnya diambil dari data suatu industri kertas di Tangerang.

Kasus 2.

Misalkan data pesanan yang masuk seperti pada tabel 4.

**Tabel 4.** Data Kasus 2

Rol pesanan	Berat pesanan
916 mm	5650 kg
880 mm	5980 kg

Data di atas dikonversi dalam jumlah rol yaitu masing-masing permintaan rol dengan lebar 916 mm dan 880 mm adalah 10 rol dan 11 rol, dengan lebar rol jumbo yaitu 2720 mm dan berat rol jumbo 1692.384 kg.

Konversi berat ke rol tersebut menggunakan rumus:

$$n_i = \frac{\text{berat rol } i}{\text{lebar rol } i} \times \frac{\text{lebar rol jumbo}}{\text{berat rol jumbo}} \quad (13)$$

Dari perhitungan Matlab diketahui bahwa pola dan jumlah rol yang digunakan dapat dilihat pada tabel 5.

**Tabel 5.** Hasil dari Kasus 2 dengan Metode Penghasil Kolom

Pola	Lebar rol pesanan		Jumlah rol
	916 mm	880 mm	
1	2	1	5
2	0	3	2

Sehingga jika hasil tersebut dikonversikan ke dalam berat maka dapat dilihat pada tabel 6.

**Tabel 6.** Hasil Konversi Berat dari Kasus 2 dengan Metode Penghasil Kolom

Pola	Lebar rol pesanan		Jumlah rol	Jumlah Berat
	916mm	880mm		
1	2	1	5	8437,032 kg
2	0	3	2	3285,216 kg

Kasus 3.

Misalkan data pesanan yang masuk seperti tabel 7.

**Tabel 7.** Data Kasus 3

Rol pesanan	Berat pesanan
1040 mm	16640 kg
880 mm	6340 kg
640 mm	5120 kg

Data di atas dikonversi dalam jumlah rol yaitu masing-masing permintaan rol dengan lebar 916 mm, 880 mm dan 640 mm adalah 20 rol, 9 rol dan 10 rol, dengan lebar rol jumbo yaitu 2750 mm dan berat rol jumbo 2200 kg.

Dari perhitungan Matlab diketahui bahwa pola dan jumlah rol yang digunakan dapat dilihat pada tabel 8.

**Tabel 8.** Hasil dari Kasus 3 dengan Metode Penghasil Kolom

Pola	Lebar rol pesanan			Jumlah rol	Lebar Rol Sisa
	1040 mm	880 mm	640 mm		
1	2	0	1	10	300 mm
2	0	3	0	3	110 mm

Sehingga jika hasil tersebut dikonversikan ke dalam berat maka dapat dilihat pada tabel 9.

**Tabel 9.** Hasil Konversi Berat dari Kasus 3 dengan Metode Penghasil Kolom

Pola	Lebar rol pesanan			Jumlah rol	Jumlah Berat	Berat Sisa
	1040 mm	880 mm	640 mm			
1	2	0	1	10	21760 kg	240 kg
2	0	3	0	3	6336 kg	264 kg

## 7. Kesimpulan

Dari kasus yang diproses didapatkan hasil yang optimal karena memberikan jumlah rol yang minimum. Terlihat bahwa dengan metode penghasil kolom menghasilkan solusi optimal untuk masalah pemotongan kertas yaitu menghasilkan jumlah rol yang optimal. Metode ini juga lebih efektif karena tidak perlu membuat semua kemungkinan pola. Pada contoh kasus 1 menghasilkan solusi yang sama dengan contoh tanpa harus membuat semua kemungkinan pola pemotongan. Sedangkan untuk masalah nyata di industri kertas masih menggunakan cara manual yaitu dengan membuat semua kemungkinan pola untuk mendapatkan pola yang menghasilkan jumlah rol yang minimum. Dengan demikian, metode ini dapat digunakan di industri kertas dalam masalah pemotongan rol kertas terutama karena metode ini lebih efektif daripada harus mencari semua kemungkinan pola.

Pada artikel ini juga terdapat aplikasi yang dibuat dengan MATLAB untuk menyelesaikan masalah pemotongan berdasarkan metode penghasil kolom dimana hasil dapat berupa jumlah rol dan berat. Aplikasi ini dibuat agar saat memasukkan data atau menampilkan data tidak perlu membuka MATLAB. Aplikasi ini dapat memproses data dari File Excel sehingga perhitungan menjadi lebih mudah.

## Daftar Pustaka

- [1] B. Susanta. *Program Linear*. Jakarta: Depdikbud. 1996.
- [2] B. Harsanto, MM. *Naskah Tutorial QM for Windows*. Bandung. 2011.
- [3] J. Bisschop. *A Cutting Stock Problem*. AIMMS Optimization Modeling. AIIMS B.V. 2016; 235-244.
- [4] K. B. Parmar, dkk. *Cutting Stock Problem: A Survey of Evolutionary Computing Based Solution in 2014 International Conference on Green Computing Communication and Electrical Engineering*, 2014.
- [5] J. Desrosiers, Marco E. Lubbecke. *A Primer in Column Generation*. Springer. 2005.
- [6] M. E. Lubbecke. *Column Generation*. Wiley Encyclopedia of Operations Research and Management Science (EORMS). 2010.

- [7] M. E. Lubbecke, Jacques Desrosiers. *Selected Topics in Column Generation*. Operations Research. Informs. 2005.
- [8] V. Chvatal. *Linear programming*. New York: W. H. Freeman and Company. 1983.