

Simulation Cutting Wood with Full Tapper Cant Sawing Method

Bezirgen Bokayev
Informatika, Fakultas MIPA
Universitas Sebelas Maret
Jl. Ir. Sutami No. 36 A Surakarta
bezza2775@student.uns.ac.id

Sarngadi Palgunadi
Informatika, Fakultas MIPA
Universitas Sebelas Maret
Jl. Ir. Sutami No. 36 A Surakarta
palgunadi@staff.uns.ac.id

Bambang Harjito
Informatika, Fakultas MIPA
Universitas Sebelas Maret
Jl. Ir. Sutami No. 36 A Surakarta
bambang_harjito@staff.uns.ac.id

ABSTRACT

Wood is the main source of raw materials required by a lumber mill. The initial process of the processing woof is the cutting of the log into boards. Problems commonly encountered wood factory is a factory uses raw materials inefficiently. This is due to limited knowledge and information in the processing of wood such as wood cutting techniques.

This study used a Greedy algorithms Integer knapsack with Full Taper Cant Sawing method, for simulation sawmill, timber size small radius 6cm, 8cm, 10cm large radius 8cm, 10cm, 12cm used in this study.

A measure of cant be limited to the width of 8cm, 10cm, 12cm thickness 6cm, 8cm, 10cm while the size of the resulting boards are 3-5cm wide and 2-3cm thick.

The simulation results show that the results are more optimal sawmill.

Kata Kunci: *Simulasi, Full Taper, Cant Sawing, Algorithma Greedy Integer Knapsack, Optimasi perolehan papan*

1. PENDAHULUAN

Indonesia merupakan salah satu negara yang mempunyai wilayah hutan yang luas. Pasokan kayu yang dihasilkan dari hutan cukup berlimpah sehingga di Indonesia banyak berdiri pabrik kayu. Pada umumnya pabrik kayu tidak langsung menggunakan kayu dalam bentuk gelondongan, namun lebih dahulu mengkonversikannya menjadi kayu berbentuk papan. Papan – papan kayu ini akan memiliki ukuran dan jumlah yang berbeda – beda, sesuai dengan kebutuhan dan permintaan industri. Pada tahun 2012, kayu gelondongan dari Indonesia, secara global meduduki peringkat 10 besar, dilihat dari volume produksi yang dihasilkan [1].

Tingkat efisiensi pada pabrik pemotongan kayu dipengaruhi oleh praktek penggergajiannya, yang berpengaruh pada volume kayu dan kualitas kayu. Metode penggergajian kayu adalah urutan pemotongan kayu dari kayu gelondongan menjadi papan [2]. Ada beberapa metode pemotongan kayu yang umumnya digunakan, diantaranya adalah cant sawing dan live sawing. Cant sawing adalah dimana kayu gelondongan diproduksi pada *headrig* atau *resaw* / mesin potong kayu dan potongan cant diproses secara bersama-sama. Metode ini lebih fokus pada jumlah produksi papan cant yang bisa dihasilkan daripada kualitas papan. Live sawing melibatkan pemotongan dengan *sawline* paralel. Para peneliti mengklaim bahwa live sawing menghasilkan serbuk gergaji yang lebih sedikit daripada cant sawing sehingga potensi untuk pemulihan kayu lebih

tinggi [3]. Peneliti cant sawing berpendapat bahwa pada live sawing untuk mendapatkan papan kayu ada bagian tepi / pinggiran kayu yang terbuang sedangkan pada cant sawing dari pinggiran kayu masih bisa dihasilkan potongan papan kayu pendek, sehingga jumlah papan yang dihasilkan lebih maksimal.

Pola penggergajian kayu untuk potongan papan menggunakan full taper dan split taper [4]. Dari penelitian [3] menyatakan bahwa full taper menjanjikan hasil yang lebih tinggi, karena kemungkinan tambahan potongan kayu dari kayu log hasilnya akan lebih baik jika semua papan dimasukkan melalui satu sisi gergaji daripada dibagi menjadi dua sisi yang berlawanan.

Untuk hasil pemotongan kayu yang optimal harus diukur sesuai radius kecil dan besar serta ukuran panjangnya. Kendalanya adalah tidak semua pabrik mengetahui penggunaan Algorithma Cant Sawing dalam pemotongan kayu tersebut. Maka penelitian ini dilakukan untuk membuat simulasi pemotongan papan kayu dengan metode Full Taper Cant Sawing untuk memperoleh hasil yang optimal.

2. ALGORITMA GREEDY

Algorithma Greedy merupakan algoritma yang lazim untuk memecahkan persoalan optimasi meskipun hasilnya tidak selalu merupakan solusi yang optimum. Sesuai arti harafiah, Greedy berarti tamak. Prinsip utama dari algoritma ini adalah mengambil sebanyak mungkin apa yang dapat peroleh sekarang [5]. Untuk memecahkan persoalan dengan algoritma Greedy, kita memurlukan elemen-elemen sebagai berikut.

- Himpunan Kandidat (C)
Himpunan ini elemen-elemen pembentuk solusi.
- Himpunan Solusi (S)
Himpunan ini berisi kandidat yang terpilih sebagai solusi persoalan. Dengan kata lain himpunan solusi adalah himpunan bagian dari himpunan kandidat.
- Fungsi Seleksi
Fungsi seleksi merupakan fungsi yang ada pada setiap langkah memilih kandidat yang paling memungkinkan guna mencapai solusi optimal.
- Fungsi kelayakan (Feasible)
Kungsi kelayakan adalah fungsi yang memeriksa apakah suatu kandidat yang telah dipilih dapat memberikan solusi yang layak dan tidak melanggar batasan atau *constraints* yang ada.
- Fungsi Objektif
Fungsi objektif adalah fungsi yang memaksimalkan atau meminimalkan solusi.

Skema umum algoritma Greedy adalah sebagai berikut.

- Inisialisasi S dengan kosong.
- Pilih sebuah kandidat C dengan fungsi seleksi.
- Kurangi C dengan kandidat yang sudah dipilih dari langkah (b) di atas.
- Periksa apakah kandidat yang dipilih tersebut bersama-sama dengan himpunan solusi membentuk solusi yang layak atau deasible (dengan fungsi kelayakan).
- Periksa apakah himpunan solusi yang lengkap serta optimal (dengan fungsi objektif).

Secara umum rumus dari algoritma Greedy adalah:

Berat barang total = $\sum_{j \in N} W_j X_j \leq K$ dimana $X_j \in \{0,1\}$

Volume barang total = $\sum_{j \in N} V_j X_j \leq K$ dimana $X_j \in \{0,1\}$

Keuntungan total = $\sum_{j \in N} P_j X_j$ dimana $X_j \in \{0,1\}$

W_j adalah berat barang, V_j adalah volume barang, P_j adalah keuntungan barang X_j adalah bernilai 0 jika barang tersebut tidak dipilih untuk dimasukkan ke dalam kapasitas Knapsack dan 1 jika barang tersebut terpilih untuk dimasukkan ke dalam kapasitas Knapsack, dan K adalah kapasitas media pengiriman.

Pada penyelesaian masalah Knapsack dengan menggunakan algoritma Greedy dapat 3 cara sebagai berikut [6].

- Greedy by Profit, pilih benda-benda dengan keuntungan maksimum dan benda-benda tersebut memiliki berat yang masih dapat ditampung oleh sisa kapasitas Knapsack.
- Greedy by Weight, pilih benda-benda dengan berat minimum dan benda-benda tersebut memiliki volume yang masih dapat ditampung oleh sisa kapasitas Knapsack.
- Greedy by volume, pilih benda-benda dengan volume minimum dan benda-benda tersebut memiliki berat yang masih dapat ditampung oleh sisa kapasitas Knapsack.
- Greedy by Weight Density, pilih benda-benda dengan keuntungan perberat yang nilainya maksimum dan benda-benda tersebut memiliki berat dan volume yang masih dapat ditampung oleh sisa kapasitas Knapsack. Rumus untuk mendapatkan density adalah:

$$D_j = \frac{P_j}{W_j} \quad (1)$$

- Greedy by Volume Density, pilih benda-benda dengan keuntungan per volume yang nilainya maksimum dan benda-benda tersebut memiliki berat dan volume yang masih dapat ditampung oleh sisa kapasitas Knapsack. Rumus untuk mendapatkan density adalah:

$$D_j = \frac{P_j}{V_j} \quad (2)$$

Pada Sebagian masalah algoritma Greedy tidak selalu berhasil memberikan solusi yang optimal. Jika jawaban terbaik mutlak tidak diperlukan, maka algoritma Greedy sering berguna untuk menghasilkan solusi hampiran (approximation), dibandingkan dengan menggunakan algoritma yang lebih rumit untuk menghasilkan solusi yang eksak. Bila algoritma Greedy optimum, maka keoptimalannya itu dapat dibuktikan secara matematis.

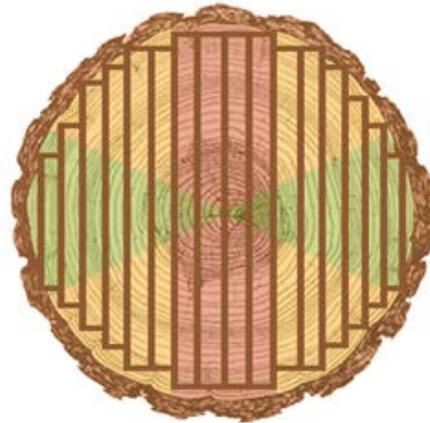
3. Sawing Method

Terdapat empat metode penggergajian kayu atau Sawing Method yaitu :

1) Live Sawing

Penggergajian yang dilakukan secara berurutan dalam satu arah atau satu bidang yang sama [3]. Pola ini seperti ilustrasi Gambar 1 baik diterapkan pada kayu yang berukuran/berdiameter kecil, pada batang yang masih muda. Keuntungannya adalah memperoleh lebar kayu yang maksimum. Kerugiannya adalah :

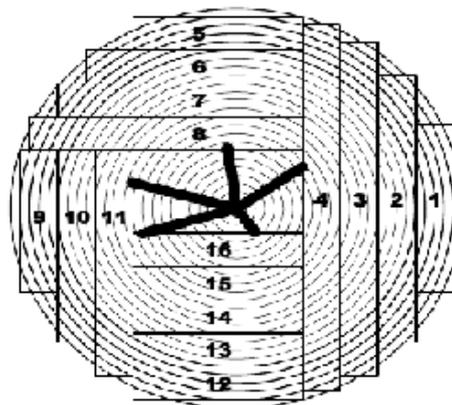
- Kayu gergajian yang mengandung pusat batang, apabila kering akan pecah memanjang pada pusat batangnya
- Kayu gergajian yang diperoleh akan melengkung, ujung-ujungnya menjauhi pusat batang.



Gambar 1. Live sawing [8]

2) Sawing Around

Gambar 2 merupakan ilustrasi Sawing Around dimana pemotongan dilakukan dengan membuang papan bagian terluar kayu terlebih dahulu. Selanjutnya secara terus-menerus kayu di putar 90o saat pemotongan berlangsung [9].



Gambar 2 Sawing around [10]

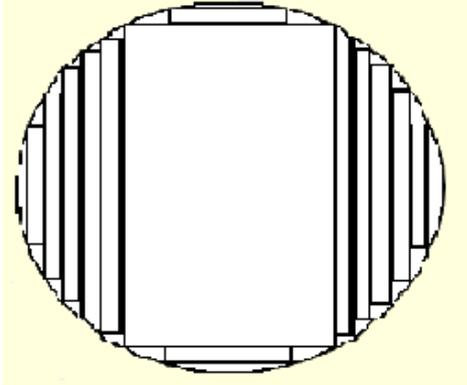
3) Cant Sawing

Pola Cant Sawing ditunjukkan seperti Gambar 3 dimana sebuah papan besar dapat mencakup satu bidang yang disebut cant [3].

Urutan penggergajian *cant sawing* secara umum adalah sebagai berikut [11]:

- Mencari dimensi terbesar yang dapat dihasilkan oleh kayu yang di sebut *cant*.

- ii. Sisi samping disekitar bagian papan *cant*, dipotong berurutan dalam satu garis sejajar untuk dimensi kayu yang lebih kecil.
- iii. Bagian atas dan bawah dari papan *cant* juga digunakan kembali. Pada pemotongan bagian ini, batang kayu diputar sampai papan *cant* yang semula dalam posisi vertical menjadi posisi horizontal.



Gambar 3. Cant sawing [12]

4) **Quarter Sawing**

Quarter saw merupakan permukaan terluas pada bidang radial kayu. Bidang radial adalah tampilan permukaan kayu yang dilihat dari orientasi sel dan tipe sel yang nampak pada sekitaran jari - jari lingkaran kayu. Jika menginginkan model *quarter saw* seperti Gambar 4 maka kayu dipotong dengan orientasi 45° – 90° terhadap bagian permukaan kayu terbesar sesuai posisi cincin tahunan kayu [9]. Penggergajian kayu gelondongan yang tegak lurus terhadap diameter kayu [13].

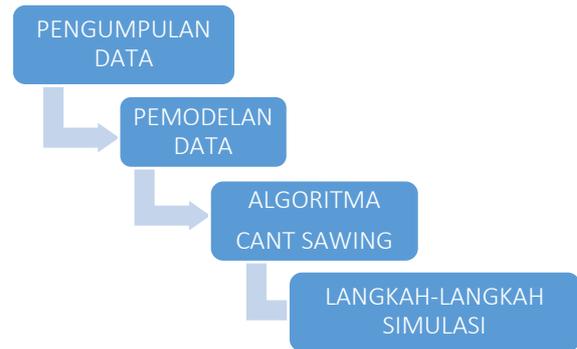


Gambar 4. Quarter sawing [14]

4. METEDOLOGI

Langkah-langkah

Langkah-langkah seperti dilihat pada gambar 5



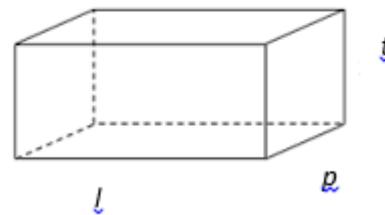
Gambar 5. Langkah-langkah

4.1. **Pengumpulan Data**

Data yang digunakan dalam penelitian berupa data ukuran diameter kayu gelondongan dan data ukuran papan kayu yang akan dipotong. Data diperoleh dengan wawancara narasumber yaitu Bapak Surono, pemilik dari Sragen Furniture Ronggo Jati Karangjati, Kalijambe.

4.2. **Pemodelan**

Simulasi yang dirancang akan menentukan nilai optimum batang kayu dengan jari-jari r yang dapat diperoleh dalam dimensi balok (panjang p), lebar (l) dan tebal atau tinggi (t). Batasan nilai optimum luas penampang *cant* (L) = $l*t$, dimana nilai maximum l dan $t = r\sqrt{2}$. Dan nilai maximum panjang balok sesuai dengan panjang kayu.



Gambar 6. Balok

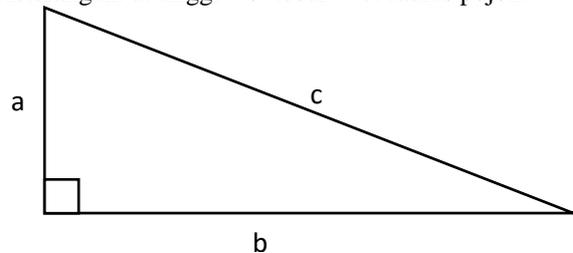
a. **Rumusan Matematis**

Rumusan matematis yang digunakan untuk menemukan radius pojok cant dalam penelitian ini adalah dengan menggunakan prinsip pythagoras dan perbandingan luas segitiga seperti dibawah ini :

Algoritma Pythagoras seperti ilustrasi Gambar 6 dan 7

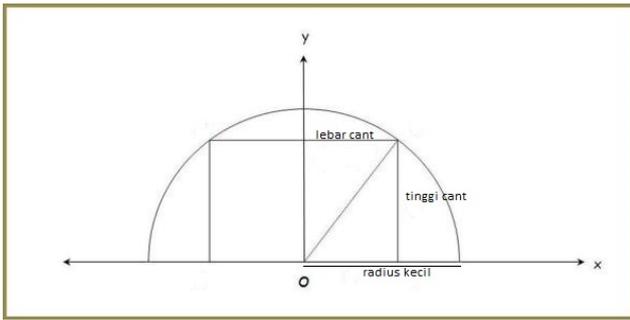
$$c = \sqrt{a^2 + b^2} \tag{3}$$

Keterangan: a: tinggi b: lebar c: radius pojok.



Gambar 7. segitiga siku-siku

Berdasarkan algoritma di atas, maka algoritma cant sawing dapat dicari menggunakan setengah lingkaran seperti gambar 7.

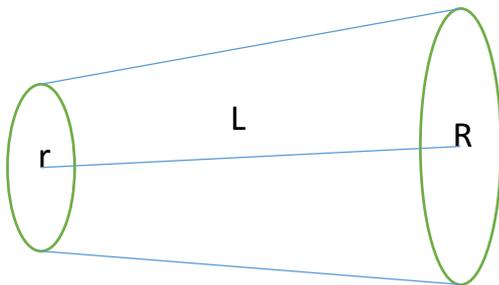


Gambar 8. Ilustrasi setengah lingkaran sebagai setengah luas penampang kayu.

4.3. Algoritma Cant Sawing

Algoritma Cant Sawing:

Input : r=Jari-jari kecil, R=jari-jari besar, L=panjang Gelondong. Seperti dilihat pada gambar 9



Gambar 9 Ukuran input-input

Output: berbagai ukuran papan dan cant yang diperoleh :

1. Hitung Ukuran Cant

$$\text{Lebar}_{\max} = \sqrt{r^2 - (0.5 * \text{tinggi cant})^2} * 2$$

2. Hitung ukuran papan pada slab 1 dan 3. Karena ukuran slab 1 dan 3 sama.

$$\text{lebar}_{\max} = \sqrt{r^2 - \left(\frac{1}{2}t_{cant} + t_{papan 1} + t_{papan 2} + \dots + t_{papan n}\right)^2} * 2$$

$$r_{sudut} = \sqrt{\left(\frac{1}{2}t_{cant} + t_{papan 1} + t_{papan 2} + \dots + t_{papan n}\right)^2 + \frac{1}{2}l_{papan n}^2}$$

$$p_{papan-n} = \frac{(R-r_{sudut})}{(R-r)/p_{gelondong}}$$

3. Hitung ukuran papan di slab 2 dan 4. . Karena ukuran slab 2 dan 4 sama. Hitungannya sama seperti slab 1 dan 3.

4. Hitung total Volume papan yang diperoleh, hitung volume gelondong, hitung persentase volume papan dibanding volume gelondong.

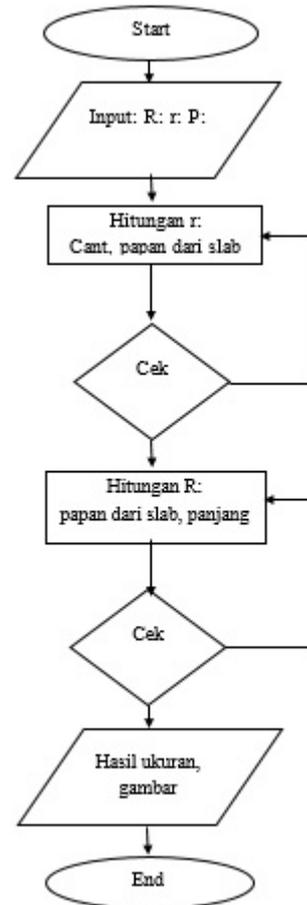
$$\text{Volume papan} = (\text{Panjang} * \text{Lebar} * \text{Tinggi})$$

$$\text{Volume Gelondong} = \left(\frac{1}{3} \pi\right) (\text{panjang gelondongan}) (r^2 + r * R + R^2)$$

4.4. Langkah-langkah Simulasi

Langkah-langkahnya adalah seperti dilihat pada gambar 10.

1. Mencari papan Cant
2. Mencari papan di jari-jari kecil
3. Mencari papan di jari-jari besar dan papan cari panjang papan.



Gambar 10. Flowchart langkah-langkah simulasi

5. PEMBAHASAN

5.1 Pembahasan Simulasi

5.1.1 Data dari bapak Surono gelondongan pada dilihat di Tabel 1.

Tabel 1 Data gelondongan

	Radius Besar	Radius kecil	Panjang
1	8	6	100
2	10	8	100
3	12	10	100
4	8	6	150
5	10	8	150
6	12	10	150
7	8	6	200
8	10	8	200
9	12	10	200

5.1.2 Awalnya harus mengetahui ukuran gelondongan. Ukuran gelondongan adalah 3 jenis kayak di tabel yaitu seperti dilihat pada Gambar 11:

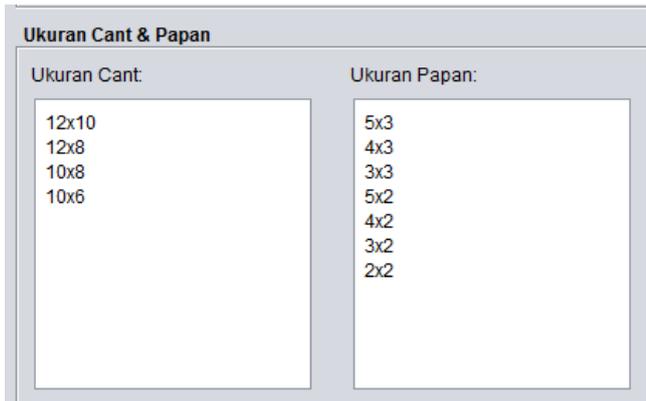
- Radius besar atau radius belakang.
- Radius kecil atau radius depan.
- Papan gelondongan.



Gambar 11. Input ukuran gelondongan

Namun memasukkan ukuran yang di batasi seperti di batasan masalah.

5.1.3 Ukuran gelondongan adalah ukuran kayu yang akan dihitung. Ukuran papan kayu ada dua macam. Ukuran cant dan ukuran papan seperti dilihat pada Gambar 12.



Gambar 12. Ukuran cant dan papan (dalam cm)

5.1.4 Di atas sudah dijelaskan bahwa selama di dalam radius kecil ukuran papan mulai di hitung dari besar ke kecil. Tetapi jika di luar radius kecil berarti harus diprioritaskan ukuran papan seperti dilihat pada gambar 13.



Gambar 13. Prioritas ukuran dan papan Setelah semuanya dimasukkan tekan tombol proceed.

5.2 Testing

Contoh Testing dengan data 5.

1.2.1. Input

Memasukkan ukuran radius besar: 10
 Memasukkan ukuran radius kecil: 8
 Memasukkan ukuran panjang gelondongan 150

1.2.2. Awalnya dicari ukuran cant

$$\text{lebar_max} = \sqrt{r^2 - (0.5 * \text{tinggi cant})^2} * 2$$

$$= \sqrt{8^2 - (0.5 * 10)^2} * 2$$

$$= 12.48$$

Apakah 12.48 > 12 kalau iya ukuran Lebar maksimum adalah 12. (TRUE)

Jadi ukuran cant adalah lebar = 12 tinggi 10
 Kalau hasil < 12 maka harus mencari ukuran yang 10.
 Kalau hasil < 10 maka harus mencari ukuran yang 8.

Setelah mendapat cant maka harus mencari ukuran papan dari slab 1 dan 2.

Karena slab 1 dan 3 ukurannya sama, slab 2 dan 4 juga ukurannya sama. Awalnya harus mencari ukuran papan yang di dalam radius kecil supaya mendapat panjang. Kalau tidak cukup baru mulai mencari di radius besar.

Untuk mencari ukuran papan mulai dari slab 1

1.2.3. Mencari ukuran Slab 1 dan 3.

a. Papan 1

$$\text{lebar_max} = \sqrt{r^2 - (\frac{1}{2}t_{cant} + t_{papan1})^2} * 2$$

$$= \sqrt{8^2 - (\frac{1}{2}10 + 3)^2} * 2$$

$$= 0$$

Kalau hasil hitungan 0 berarti tinggi papan tidak mencukupi.

0 < 5 berarti ini salah. Tinggi papan lebih kecil dari 3. Kalau ukuran 3 tidak cukup akan dicari dengan ukuran tinggi 2.

$$\text{lebar_max} = \sqrt{r^2 - (\frac{1}{2}t_{cant} + t_{papan1})^2} * 2$$

$$= \sqrt{8^2 - (\frac{1}{2}10 + 2)^2} * 2$$

$$= 7.74$$

Apakah 7.74 > 5. Kalau iya (TRUE)

Jadi lebar papan1=5 tinggi papan1= 2.

b. Papan 2

Radius kecil adalah 8. Maka 5 termasuk ke cant + 2 untuk papan1. Jadi sisa radius kecil adalah 1. Satu tidak cukup untuk papan kayu. Jadi papan 2 akan mulai dihitung dengan radius besar. Tetapi untuk hitungan radius besar ada dua pilihan. Perhitungan ukuran besar ke kecil atau kecil ke besar.

$$r_{sudut} = \sqrt{(\frac{1}{2}t_{cant} + t_{papan1} + t_{papan2})^2 + (\frac{1}{2}(l_{papan2}))^2}$$

Kalau kecil ke besar:

$$r_{sudut} = \sqrt{(\frac{1}{2}10 + 2 + 2)^2 + (\frac{1}{2}2)^2}$$

$$= 9.05$$

Setelah menemukan r_sudut kita cari panjang papan.

$$\text{panjang}_{papan-n} = \frac{(R-r_{sudut})}{(R-r)/p_{gelondong}}$$

$$P_{papan2} = \frac{(10-9.055385138137417)}{(10-8)/150}$$

$$= \frac{0.9446148618625834}{0.0133333333333333}$$

$$= 70.84.$$

Jadi lebar papan 2 = 2, tinggi papan 2 = 2, Panjang papan 70.85

Di slab satu radius besar adalah 10. Maka 5 sudah di ambil untuk cant + 2 buat papan1 + 2 buat papan 2. Sisa tinggi papan adalah 1. 1 lebih kecil dari dua jadi tidak cukup buat papan.

Berarti slab 1 dan 3 menghasilkan 4 papan.

1.2.4. Mencari ukuran slab 2 dan 4

Cara hitungannya sama seperti slab 1 dan 3.

a. Papan 1

Mencari dengan ukuran tinggi 3cm:

$$\begin{aligned} \text{lebar_max} &= \sqrt{r^2 - \left(\frac{1}{2}l_{cant} + t_{papan\ 1}\right)^2} * 2 \\ &= \sqrt{8^2 - \left(\frac{1}{2}12 + 3\right)^2} * 2 \\ &= 0 \end{aligned}$$

Dari hasil perhitungan ukuran tinggi 3 tidak mencukupi.

Mencari dengan ukuran tinggi 2 :

$$\begin{aligned} \text{lebar_max} &= \sqrt{r^2 - \left(\frac{1}{2}l_{cant} + t_{papan\ 1}\right)^2} * 2 \\ &= \sqrt{8^2 - \left(\frac{1}{2}12 + 2\right)^2} * 2 \\ &= 0 \end{aligned}$$

Dari hasil perhitungan ukuran tinggi 2 tidak mencukupi.

Ternyata untuk mencari di dalam radius kecil tidak mencukupi. Jadi harus mencari di radius besar.

$$\begin{aligned} r_{sudut} &= \sqrt{\left(\frac{1}{2}l_{cant} + t_{papan\ 1}\right)^2 + \left(\frac{1}{2}l_{papan\ 2}\right)^2} \\ &= \sqrt{\left(\frac{1}{2}12 + 2\right)^2 + \left(\frac{1}{2}2\right)^2} \\ &= 8.06. \end{aligned}$$

$$\text{panjang}_{papan-n} = \frac{(R - r_{sudut})}{(R - r) / p_{gelondong}}$$

$$\text{Ppapan} = \frac{(10 - 8.06)}{(10 - 8) / 150}$$

Panjang papan 145.33

Di slab dua dan 4 radius besar adalah 10. Maka 6 sudah di ambil untuk cant +2 untuk papan 1. Sisa tinggi papan adalah 2 . Kalau sisa tinggi 2 tidak mencukupi untuk radius sudut papan 2. Jadi slab 2 ada 1 papan. Slab 4 juga jadi 1. Totalnya 2 papan dari slab 2 dan 4.

Total hasil dari R:10 r:8 P:150

Tabel 2. Hasil ukuran kayu (dalam cm)

		Lebar	Tinggi	Panjang	Volume
	Cant	12	10	150	18000
Slab 1	Papan1	5	2	150	1500
	papan2	2	2	70.85	283.4
Slab 2	Papan1	2	2	145.33	581.32
	papan2	-	-	-	
Slab 3	Papan1	5	2	150	1500
	papan2	2	2	70.85	283.4
Slab 4	Papan1	2	2	145.33	581.32
	papan2	-	-	-	

Daftar Pustaka

- Jurgensen, C., Kollert, W. & Lebedys, A., 2014. *Assessment Of Industrial Roundwood Production From Planted Forests*. 4 Ed. Rome: Food And Agriculture Organization Of The United Nations.
- Steele, P.H.1984. Factors Determining Lumber Recovery in Sawmilling. U.S.D.A. Forest Service. General technical Report FPL-39.
- Hallock, H. & Lewis, D. W., 1976. *Is There A "Best" Sawing Method?*, Wisconsin: Forest Product Laboratory.
- Denig, J.1993. Small sawmill handbook. Miller Freeman, San Francisco.
- Brassard G, 1996, Fundamentals of algorithmics, Prentice-Hall, New Jersey.
- Martello, S. & Toth, P., 1990. *Knapsack Problem : Algorithm and Computer Implementations*. England: John Wiley & Sons.
- Prof. Olivier de Weck., 2003. Multidisciplinary System Design Optimization (MSDO) Spring 2003
- http://www.hardwoodsofwisconsin.com/saw_cuts.htm
- Bond, B., 1914. Sawing Logs for Quartersawn Lumber, Tennessee: Agricultural Extension Service.
- <http://www.stephanwoodworking.com/TreesAndLumber.htm>
- Walker, J. C. F., 2006. *Primary Wood Processing Principles And Practice*. Dordrecht Nethetlands: Springer.
- <http://delphiforfun.org/Programs/Sawmill%20Patterns.htm>
- Martono, B. et al., 2008. *Teknik Perkayuan untuk SMK*. Jakarta: Departemen Pendidikan Nasional.
- <http://www.indeco.net.au/timber-species/detail.1802045153.html>
- <http://www.vincentkohler.ch/billon.html>