



UNIVERSITAS INDONESIA

**OPTIMASI PERMASALAHAN TATA LETAK MESIN
DENGAN MENGGUNAKAN METODE *CLUSTER BOUNDARY
SEARCH* DAN ALGORITMA *META-HEURISTIC***

SKRIPSI

**ARIEL
0706274464**

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK INDUSTRI
DEPOK
JUNI 2011**



UNIVERSITAS INDONESIA

**OPTIMASI PERMASALAHAN TATA LETAK MESIN
DENGAN MENGGUNAKAN METODE *CLUSTER BOUNDARY
SEARCH* DAN ALGORITMA *META-HEURISTIC***

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik

**ARIEL
0706274464**

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK INDUSTRI
DEPOK
JUNI 2011**

HALAMAN PERNYATAAN ORISINALITAS

**Skripsi ini adalah hasil karya saya sendiri,
dan semua sumber baik yang dikutip maupun yang dirujuk
telah saya nyatakan dengan benar**

Nama : Ariel

NPM : 0706274464

Tanda tangan :

Tanggal : 14 Juni 2011



HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :
Nama : Ariel
NPM : 0706274464
Program Studi : Teknik Industri
Judul Skripsi : Optimasi Permasalahan Tata Letak Mesin Dengan
Menggunakan Metode *Cluster Boundary Search*
dan Algoritma *Meta-Heuristic*


Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Industri, Fakultas Teknik, Universitas Indonesia

DEWAN PENGUJI

Pembimbing : Akhmad Hidayatno, ST., MBT (.....)

Anggota : Ir. Amar Rachman, MEIM. (.....)

Anggota : Ir. Fauzia Dianawati, M.Si. (.....)

Anggota : Ir. Dendy Ishak, MSIE. (.....)

Ditetapkan di : Depok

Tanggal : 21 Juni 2011

UCAPAN TERIMA KASIH

Puji syukur kepada Tuhan Yesus Kristus atas kasih dan karunia-Nya sehingga dengan kekuatan-Nya skripsi ini dapat selesai dengan baik untuk kemuliaan-Nya. Penulisan skripsi ini dilakukan dalam rangka melengkapi salah satu persyaratan untuk menyelesaikan Program Pendidikan Sarjana Teknik Industri, Fakultas Teknik Universitas Indonesia. Saya menyadari bahwa tanpa bantuan dan bimbingan dari berbagai pihak, sulit bagi saya untuk menyelesaikan skripsi ini. Pada kesempatan ini, saya menyampaikan ucapan terima kasih kepada:

1. Bapak Komarudin ST., M.Eng selaku dosen pembimbing skripsi, untuk segala bimbingan, bantuan, arahan dan dukungan dalam pengerjaan skripsi ini.
2. Bapak Akhmad Hidayatno ST., MBT selaku dosen pembimbing pengganti pak Komar yang telah memberikan saran, dan pengarahan, sehingga saya dapat maju ke siding skripsi.
3. Aziiz Sutrisno, Gersianto Bagusputra dan Daril B. Jehoiada untuk izin pemakaian SEMS untuk pengerjaan skripsi saya.
4. Oscarses Sukoco, Gersen Samuel, Paulus Bangun, Tarida Sinambela dan Maolana Hakim atas semua bantuan yang telah diberikan, khususnya dalam mempelajari program MATLAB dan mengisi waktu kebersamaan semasa mengerjakan skripsi di SEMS Departemen Teknik Industri Universitas Indonesia
5. Mas Latif dan Babe yang direpotkan untuk tidak mematikan listrik SEMS demi iterasi program yang sedang saya lakukan dalam penyusunan skripsi
6. Orang tua, Papa, Mama, Adik, serta keluarga besar yang juga turut mendoakan dan mendukung saya.
7. Noviandrianus Sunairdnaivon dan Adi Suryadinata yang memberikan saran, masukan dan memotivasi saya untuk menyelesaikan skripsi ini.
8. Teman-teman SMAK3 Penabur, a2g dan Abbalove Ministries yang tidak bisa disebut satu per satu atas pengertian dan dukungannya dalam pengerjaan skripsi ini.

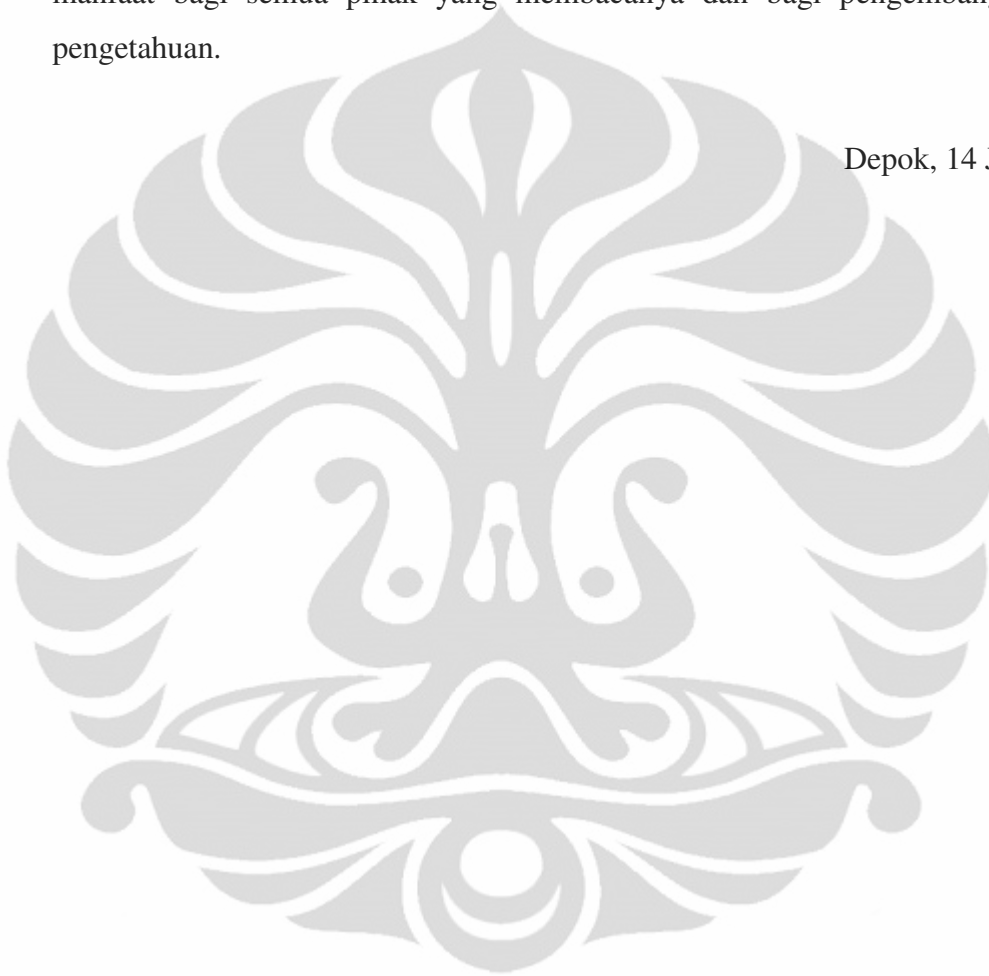
9. Seluruh teman-teman TI07 Universitas Indonesia yang telah berjuang bersama dan memberikan semangat yang luar biasa.

10. Semua pihak yang terlibat dan telah membantu penulis sehingga skripsi ini dapat terselesaikan dengan baik.

Akhirnya, penulis berharap agar skripsi ini bisa memberikan inspirasi dan manfaat bagi semua pihak yang membacanya dan bagi pengembangan ilmu pengetahuan.

Depok, 14 Juni 2011

Penulis



**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama : Ariel
NPM : 0706274464
Program Studi : Teknik Industri
Departemen : Teknik Industri
Fakultas : Teknik
Jenis Karya : Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty-Free Right*)** atas karya ilmiah saya yang berjudul:

**Optimasi Permasalahan Tata Letak Mesin dengan menggunakan Metode
Cluster Boundary Search dan algoritma *Meta-Heuristic***

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalih media/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta. Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Jakarta

Pada tanggal : 14 Juni 2011

Yang Menyatakan

(Ariel)

ABSTRAK

Nama : Ariel
Program Studi : Teknik Industri
Judul : Optimasi Permasalahan Tata Letak Mesin dengan menggunakan Metode *Cluster Boundary Search* dan algoritma *Meta-Heuristic*

Berbagai macam metodologi dan algoritma telah dikembangkan untuk memecahkan permasalahan tata letak mesin (MLP). Dalam penelitian ini, metode *Cluster Boundary Search* (CBS), algoritma *Simulated Annealing* (SA) and *Tabu Search* (TS) dikombinasikan dengan tujuan untuk mendapatkan hasil optimasi MLP yang lebih baik. Metode CBS melakukan proses optimasi dengan meletakkan sebuah mesin terpilih di sekeliling batas luar mesin yang sudah diletakkan. Mesin pertama yang akan dikelilingi dipilih berdasarkan konektivitas terbesar dengan seluruh mesin. Mesin yang belum diletakkan secara berurutan diletakkan pada posisi optimumnya pada batas keliling dari mesin yang sudah diletakkan sampai semua mesin sudah diletakkan. Hasil dari metode CBS selanjutnya dikombinasikan dengan SA dan TS dengan tujuan untuk mendapatkan *layout* tata letak yang paling optimal.

Kata kunci:

Permasalahan Tata Letak Mesin, Metode *Cluster Boundary Search*, *Simulated Annealing*, *Tabu Search*, Konektivitas Terbesar dengan Seluruh Mesin

ABSTRACT

Name : Ariel
Study Program : Industrial Engineering
Title : Machine Layout Problem Optimization using Cluster Boundary Search Method and Meta-Heuristic algorithm

Various kinds of methodologies and algorithms have been developed to solve Machine Layout Problem (MLP). In this paper, Cluster Boundary Search (CBS) method, Simulated Annealing (SA) and Tabu Search (TS) algorithms are combined with the aim of obtaining better MLP optimization result. CBS method conducts its optimization process by placing a selected unplaced machine singly along a boundary line around the cluster of the placed machines. The first machine to be circumambulated is selected on the basis of the highest value of connectivity with all the machines. The unplaced machines are sequentially located to their optimum position on the formed boundary of the placed machines until all the machines are placed. The result of BCS method is furthermore combined with SA and TS in order to obtain the most optimal facilities layout.

Keywords:

Machine Layout Problem, Cluster Boundary Search Method, Simulated Annealing, Tabu Search, Highest value of connectivity with all the machines

DAFTAR ISI

HALAMAN PERNYATAAN ORISINALITAS	ii
HALAMAN PENGESAHAN	iii
UCAPAN TERIMA KASIH	iv
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI.....	vi
ABSTRAK	vii
DAFTAR ISI.....	ix
DAFTAR TABEL	xi
DAFTAR GAMBAR.....	xii
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Diagram Keterkaitan Masalah	2
1.3 Rumusan Permasalahan	2
1.4 Tujuan Penelitian	3
1.4 Ruang Lingkup Penelitian	4
1.5 Metodologi Penelitian	4
1.6 Sistematika Penulisan	6
BAB 2 LANDASAN TEORI	9
2.1 Machine Layout Problem	9
2.1.1 Definisi	9
2.1.2 <i>Facility Layout Problem</i> / Permasalahan Tata Letak Fasilitas.....	9
2.1.3 Metode Penyelesaian MLP	10
2.2 Metode <i>Cluster Boundary Search</i>	12
2.3 Algoritma Simulated Annealing	17
2.3.1 Konsep Dasar <i>Simulated Annealing</i>	17
2.3.2 Langkah-langkah <i>Simulated Annealing</i>	18
2.4 Algoritma Tabu Search	19
2.4.1 Pengertian <i>Tabu Search</i>	19
2.4.2 Langkah-langkah <i>Tabu Search</i>	19
BAB 3 PENGUMPULAN DATA.....	21
BAB 4 PENGOLAHAN DATA DAN ANALISA	26
4.1 Pengolahan Data	26
4.1.1 Perancangan Model dan Algoritma Optimasi	26
4.1.1.1 Algoritma Fungsi Objektif	26
4.1.1.2 Pengurutan Penempatan Mesin dengan Konektivitas Terbesar ..	28
4.1.1.3 Perancangan Model Koordinat <i>Output</i>	28
4.1.1.4 Improvisasi <i>Layout</i>	33
4.1.1.5 Optimasi dengan <i>Simulated Annealing</i>	33
4.1.1.6 Optimasi dengan <i>Tabu Search</i>	35
4.1.2 Verifikasi dan Validasi Program.....	37
4.2 Pengolahan Data dan Hasil.....	41
4.3 Analisa Pengolahan Data.....	46
4.3.1 Analisa Hasil	46
4.3.2 Analisa Waktu Pengoperasian Program.....	50

4.3.3	Analisa Metode <i>Cluster Boundary Search</i> dan Algoritma <i>Meta-Heuristic</i>	52
BAB 5	KESIMPULAN	56
REFERENSI		57



DAFTAR TABEL

Tabel 2.1 Jenis-Jenis FLP	10
Tabel 2.2 Perhitungan Optimasi dari Gambar 2.4.....	15
Tabel 2.3 Koordinat Mesin Hasil Optimasi	15
Tabel 2.4 Analogi Sistem Fisis dengan Proses Optimasi.....	17
Tabel 3.1 Daftar Set Data.....	21
Tabel 3.2 Karakteristik Data	22
Tabel 3.3 Daftar Publikasi Yang Melakukan Pengolahan Data Terhadap Data Set	23
Tabel 3.4 Solusi Terpublikasi Penelitian Sebelumnya.....	25
Tabel 4.1 Ukuran Mesin KH4 untuk Validasi Program.....	37
Tabel 4.2 Tabel Aliran Material KH4 Validasi Program.....	38
Tabel 4.3 Parameter Validasi Program	38
Tabel 4.4 Output Koordinat Mesin Validasi Program	38
Tabel 4.5 Output Koordinat Mesin secara Manual	39
Tabel 4.6 Matriks Jarak <i>Centroid</i> Mesin untuk Validasi Program	40
Tabel 4.7 Matriks Biaya <i>Material Handling</i> untuk Validasi Program	40
Tabel 4.8 Parameter Pengolahan Data dengan <i>Simulated Annealing</i>	41
Tabel 4.9 Parameter Pengolahan Data dengan <i>Tabu Search</i>	41
Tabel 4.10 Hasil Pengolahan Data dengan <i>Simulated Annealing</i>	42
Tabel 4.11 Hasil Pengolahan Data dengan <i>Tabu Search</i>	43
Tabel 4.12 Perbandingan CBS/SA dengan Metode Lain.....	44
Tabel 4.13 Perbandingan CBS/TS dengan Metode Lain	45
Tabel 4.14 Perbandingan CBS-SA/TS dengan Metode Terdahulu pada Data Set KH.....	46
Tabel 4.15 Perbandingan CBS-SA/TS dengan Metode Terdahulu pada Data Set NUG	48
Tabel 4.16 Perbandingan CBS-SA/TS dengan Metode Terdahulu pada Data Set OPT	49
Tabel 4.17 Perbandingan CBS-SA/TS dengan Metode Terdahulu pada Data Set DUN	50

DAFTAR GAMBAR

Gambar 1.1 Diagram Keterkaitan Masalah.....	3
Gambar 1.2 Diagram Alir Metodologi Penelitian.....	7
Gambar 2.1 Tata Letak Mesin Diskrit.....	11
Gambar 2.2 Proses Pengelilingan Mesin Pertama oleh Mesin Kedua.....	13
Gambar 2.3 Grafik Nilai Fungsi Objektif Peletakan Mesin Kedua dalam Proses Keliling.....	13
Gambar 2.4 Proses Pengelilingan Mesin Pertama dan Kedua oleh Mesin Ketiga	14
Gambar 2.5 Grafik Nilai Fungsi Objektif Peletakan Mesin Ketiga dalam Proses Keliling.....	15
Gambar 4.1 Fase Awal Pembentukan <i>Boundary</i>	29
Gambar 4.2 Fase I Updating <i>Boundary</i>	29
Gambar 4.3 Fase II Updating <i>Boundary</i>	30
Gambar 4.4 Identifikasi <i>Boundary</i> Berdasarkan Koordinat.....	31
Gambar 4.5 <i>Boundary</i> Pada Keadaan Tertentu.....	32
Gambar 4.6 Hasil <i>Layout</i> MATLAB untuk Validasi Program	39
Gambar 4.7 Hasil <i>Layout</i> Perhitungan Manual.....	40
Gambar 4.8 Proses Umum Metode <i>Cluster Boundary Search</i>	53
Gambar 4.9 <i>Improvement</i> pada Metode <i>Cluster Boundary Search</i>	54

BAB 1

PENDAHULUAN

1.1 Latar Belakang Masalah

Industri mengalami perkembangan dari zaman ke zaman. Titik tolak perubahan industri global secara signifikan dimulai ketika terjadinya revolusi industri di Inggris. Revolusi industri adalah perubahan yang menyangkut dasar dan pokok-pokok kehidupan masyarakat dari kegiatan ekonomi agraris ke ekonomi industri yang menggunakan mesin dalam dalam mengolah bahan mentah menjadi produk yang siap pakai. Revolusi industri telah mengubah cara kerja manusia dari penggunaan tangan menjadi menggunakan mesin. Istilah revolusi industri diperkenalkan oleh Friedrich Engels dan Louis- Auguste Blanqui di pertengahan abad ke-19.

Mesin telah banyak mengurangi beban kerja manusia dalam melakukan proses produksi di industri-industri. Hal ini menyebabkan manusia dapat dengan lebih leluasa mengatur alur dan sistem produksi yang dilakukan oleh mesin mesin tersebut. Tata letak mesin akan dapat diatur sesuai dengan perencanaan pengelola industri. Namun dalam pengaturan tata letak mesin (*machine layout problem*) tersebut adalah penting untuk mempertimbangkan efisiensi peletakan mesin-mesin tersebut. Di sisi lain, peletakan yang efisien tentunya akan mengurangi keseluruhan biaya yang secara tidak langsung memaksimalkan keuntungan. Efisiensi yang akan dibahas dan diteliti di dalam skripsi saya adalah mengenai biaya perpindahan material (*material handling*) dari mesin satu ke mesin lainnya dan sebaliknya.

Biaya perpindahan material dalam hal ini akan difokuskan bergantung pada total jumlah mesin produksi, seberapa jauh material produksi berpindah dan jumlah perpindahan material pada mesin-mesin tersebut. Jika MLP ada pada sistem produksi dengan alur produksi sederhana yang linear dan searah, maka hal ini tidak akan menyulitkan untuk melakukan pengaturan dan penempatan mesin. Sebaliknya, jika alur produksi mesin tidak searah, yaitu adanya proses pengulangan pada mesin yang sama, jumlah perpindahan material dan ukuran setiap mesin berbeda, maka pengaturan tata letak mesin

tidaklah mudah dilakukan untuk mendapatkan solusi yang mempunyai *material handling cost* terkecil. Riset terhadap *machine layout problem* menunjukkan bahwa biaya *material handling* dapat mencapai 70 persen dari biaya keseluruhan operasional pabrik. Oleh karena itu, pengaturan tata letak mesin yang efektif dapat mengurangi biaya *material handling* menjadi di bawah 30 persen dari biaya keseluruhan operasional pabrik.

Beberapa studi optimasi tata letak mesin telah dilakukan untuk mendapatkan solusi optimal dari permasalahan ini. Penelitian yang sudah dilakukan mencakup metode perhitungan diskrit dan kontinu dengan beberapa algoritma yang sudah dikenal secara luas seperti *ant colonization*, *simulated annealing* dan *genetic algorithm*. Dalam skripsi ini, penulis mencoba melakukan optimasi tata letak mesin dengan perhitungan diskrit menggunakan metode *cluster boundary search*. Metode ini pernah digunakan sebelumnya dalam penelitian di Umm Al-Qura University yang dilakukan oleh M. Hasan imam & Mustahsan Mir. Studi optimasi yang akan saya lakukan di skripsi ini adalah dengan mengkombinasikan metode *Cluster Boundary Search* (CBS) dengan algoritma *meta-heuristic*. Hasil optimasi yang didapatkan dari penelitian ini akan dibandingkan dengan hasil dari metode yang berbeda pada permasalahan tata letak mesin yang sama.

1.2 Diagram Keterkaitan Masalah

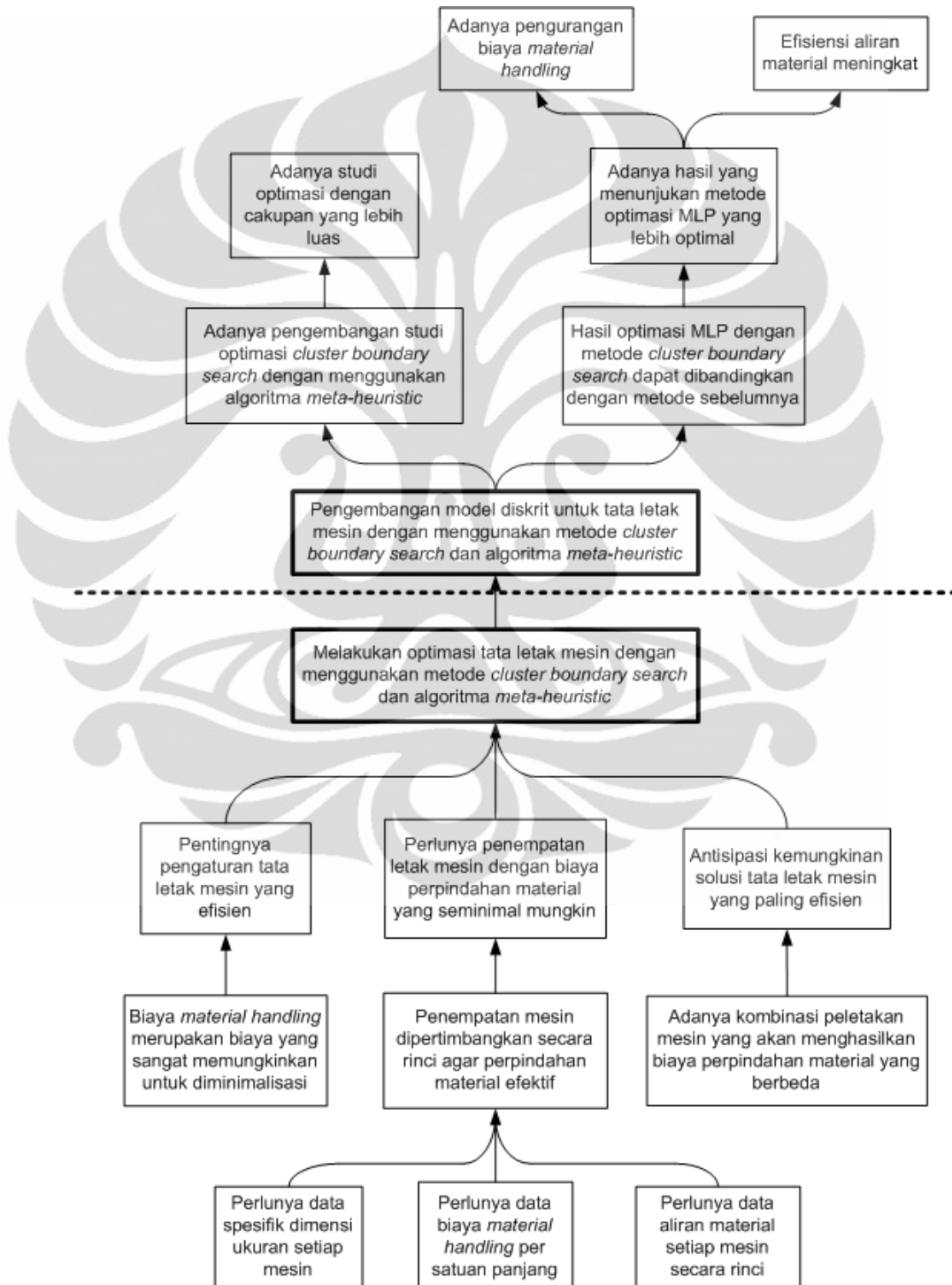
Berdasarkan latar belakang masalah di atas, saya membuat sebuah diagram keterkaitan masalah untuk memperlihatkan sebab-akibat, penyebab permasalahan dan alur pemecahan masalah. Diagram keterkaitan masalah dapat dilihat pada Gambar 1.1.

1.3 Rumusan Permasalahan

Fokus permasalahan dari penelitian ini adalah pengembangan model MATLAB yang dapat memberikan solusi optimal dari permasalahan tata letak mesin dengan menggunakan metode *cluster boundary search* dan algoritma *meta-heuristic*. Hasil yang didapatkan dari metode ini akan dibandingkan dengan metode lain pada permasalahan tata letak mesin yang sama.

1.4 Tujuan Penelitian

Berdasarkan latar belakang dan diagram keterkaitan masalah, tujuan penelitian ini adalah untuk mendapatkan hasil optimasi oleh metode *cluster boundary search* dan algoritma *meta-heuristic* untuk dibandingkan dengan hasil dari metode-metode lainnya.



Gambar 1.1 Diagram Keterkaitan Masalah

1.4 Ruang Lingkup Penelitian

Dalam penelitian ini ada pembatasan masalah dengan tujuan agar hasil penelitian yang diperoleh sesuai dengan tujuan pelaksanaannya. Adapun ruang lingkup penelitian ini adalah:

1. Penelitian ini menggunakan metode *cluster boundary search* dan algoritma *meta-heuristic*. CBS adalah salah satu metode optimasi tata letak mesin dan algoritma *meta-heuristic* digunakan untuk mencari kemungkinan solusi lain yang lebih optimal.
2. Data penelitian ini menggunakan data dari penelitian-penelitian sebelumnya yang telah dipublikasikan.
3. Hasil dari penelitian ini adalah minimalisasi biaya perpindahan material (*material handling*) tanpa memperhitungkan biaya-biaya lainnya seperti biaya material yang terlambat di-input ke mesin.
4. Luas area total peletakan mesin diasumsikan tidak terbatas, dalam artian tidak dibatasi dengan panjang dan lebar tertentu.
5. Dimensi ukuran mesin adalah persegi panjang, yaitu area terluar dari mesin dan sifatnya tetap.
6. Jarak antar mesin adalah dengan menggunakan perhitungan jarak *rectilinear*, *squared euclidean* dan *euclidean*.

1.5 Metodologi Penelitian

Metodologi penelitian yang digunakan dalam skripsi ini secara sistematis dijelaskan pada gambar 1.2 dengan uraian sebagai berikut:

1. Identifikasi permasalahan dan mencari ide solusi

Di tahap awal ini penulis melakukan *brain storming* dan mencari tema-tema yang menarik di internet, jurnal dan skripsi yang terpublikasi, untuk dijadikan objek dari penelitian di skripsi penulis.

2. Perumusan masalah

Di tahap ini penulis merumuskan beberapa permasalahan dari tema-tema yang menarik untuk dijadikan bahan penelitian. Perumusan masalah yang didapatkan di tahap ini adalah pengaturan tata letak pabrik yang efisien dan efektif.

3. Penetapan beberapa kemungkinan solusi

Untuk mendapatkan beberapa alternatif solusi dari tata letak pabrik, penulis melakukan konsultasi dengan dosen pembimbing. Dalam konsultasi ini, dosen pembimbing memberikan alternatif solusi untuk menggunakan metode *cluster boundary search* dalam mengoptimasi tata letak mesin.

4. Studi literatur

Studi ini dilakukan pada jurnal-jurnal yang terkait dengan permasalahan tata letak mesin dan fasilitas. Secara lebih spesifik penulis melakukan studi pada jurnal penelitian optimasi dengan metode *cluster boundary search*, algoritma *simulated annealing* dan *tabu search*.

5. Penentuan topik penelitian

Setelah perumusan permasalahan dan alternatif solusi telah dipelajari, topik dari skripsi penulis dapat ditentukan. Topik skripsi penulis adalah optimasi tata letak mesin dengan metode *boundary cluster search* dan algoritma *meta-heuristic*.

6. Penentuan tujuan penelitian

Dengan mengacu pada permasalahan yang ada, adapun tujuan dari penelitian ini adalah untuk memperoleh hasil optimasi dari tata letak mesin dengan jarak perpindahan material yang terpendek.

7. Pengumpulan data

Data penelitian didapatkan dari studi literatur pada jurnal-jurnal yang telah dipublikasikan sebelumnya, yang mencakup jumlah mesin, jumlah aliran material, ukuran mesin dan biaya perpindahan material.

8. Pembuatan model optimasi tata letak mesin

Dengan dasar teori dan data tata letak mesin, penulis melakukan pengembangan model optimasi ini dengan menuliskan kode-kode pemrograman metode *cluster boundary search* dan algoritma *meta-heuristic* pada perangkat lunak MATLAB R2009b.

9. Verifikasi dan pengujian program

Verifikasi ini dilakukan untuk mengetahui validitas dari pemrograman optimasi tata letak mesin ini. Pengujian juga dilakukan

pada data penelitian untuk mendapatkan tata letak mesin dari metode di skripsi ini.

10. Analisa hasil pengolahan data

Hasil dari pengolahan data dianalisis dari aspek-aspek tertentu dan kemudian dibandingkan dengan hasil optimasi dari metode-metode lainnya pada masalah tata letak mesin yang sama.

11. Membuat kesimpulan

Dalam tahap ini, penulis menyimpulkan mengenai keseluruhan penelitian yang mencakup kesimpulan dari dasar teori dalam kaitannya dengan pengolahan data, hasil analisis dan hasil dari penelitian lainnya terhadap permasalahan tata letak mesin yang sama.

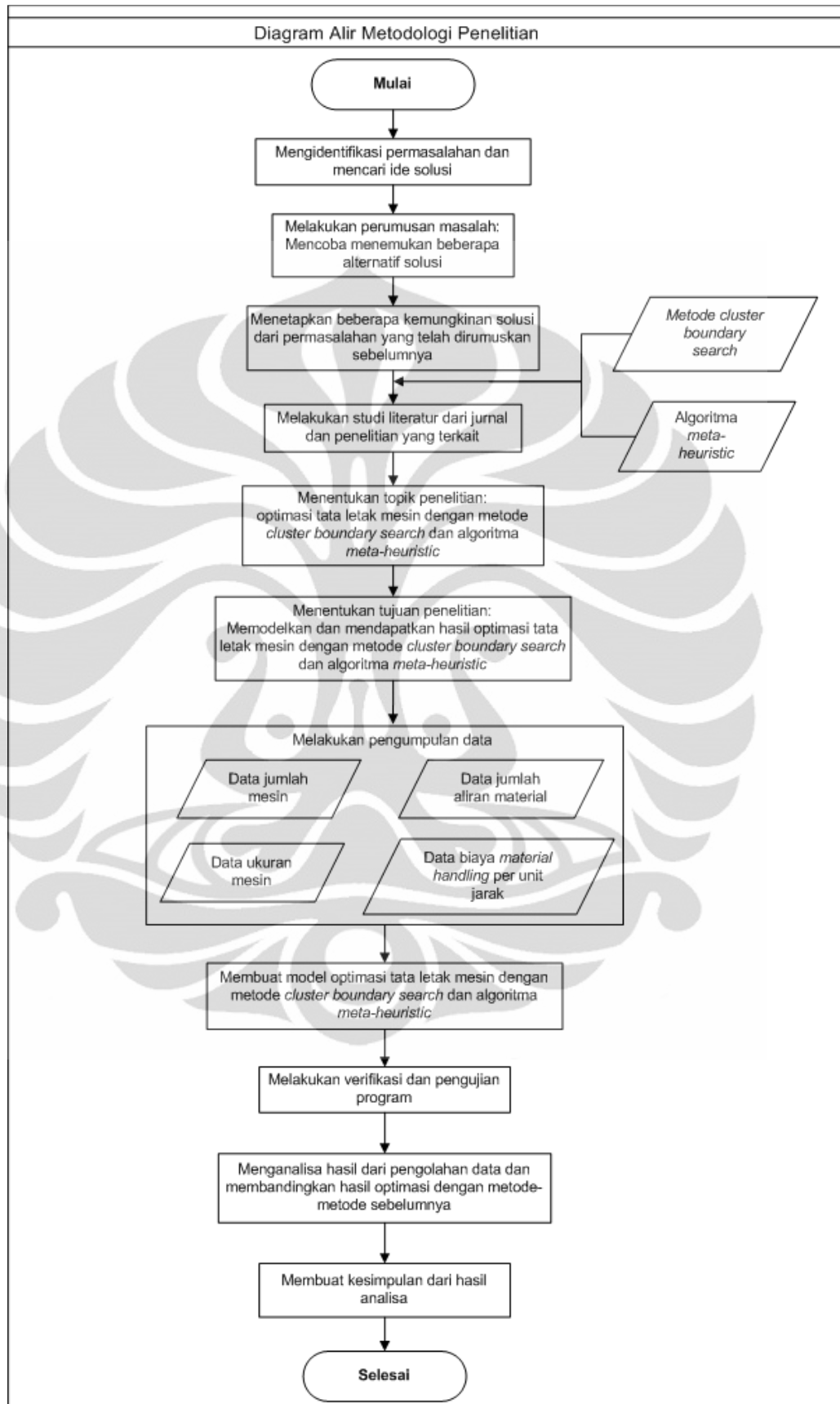
Diagram alir metodologi penelitian yang secara sistematis menggambarkan tahapan penelitian ini dapat dilihat pada Gambar 1.2.

1.6 Sistematika Penulisan

Skripsi mengenai optimasi tata letak mesin dengan metode *boundary cluster search* dan algoritma *meta-heuristic* ini dibagi menjadi beberapa bab.

Pada bab pertama penulis menjelaskan latar belakang masalah yang mendorong dilakukannya penelitian ini. Diagram keterkaitan masalah, rumusan permasalahan dan tujuan penelitian dijelaskan dalam bab ini untuk memberikan penjelasan yang lebih rinci mengenai sebab dan akibat dari penelitian ini. Ruang lingkup penelitian menjelaskan batasan-batasan asumsi awal yang menjadi parameter penelitian ini. Metodologi penelitian menjelaskan urutan dan sistematika penelitian ini secara menyeluruh.

Pada bab kedua penulis menjelaskan dasar teori yang menjadi dasar penelitian ini. Adapun dasar teori yang akan digunakan adalah teori mengenai *Machine Layout Problem (MLP)*, Metode *Cluster Boundary Search (CBS)* dan Algoritma *meta-heuristic*. Dasar teori dipelajari dari buku, jurnal dan penelitian lain yang terkait.

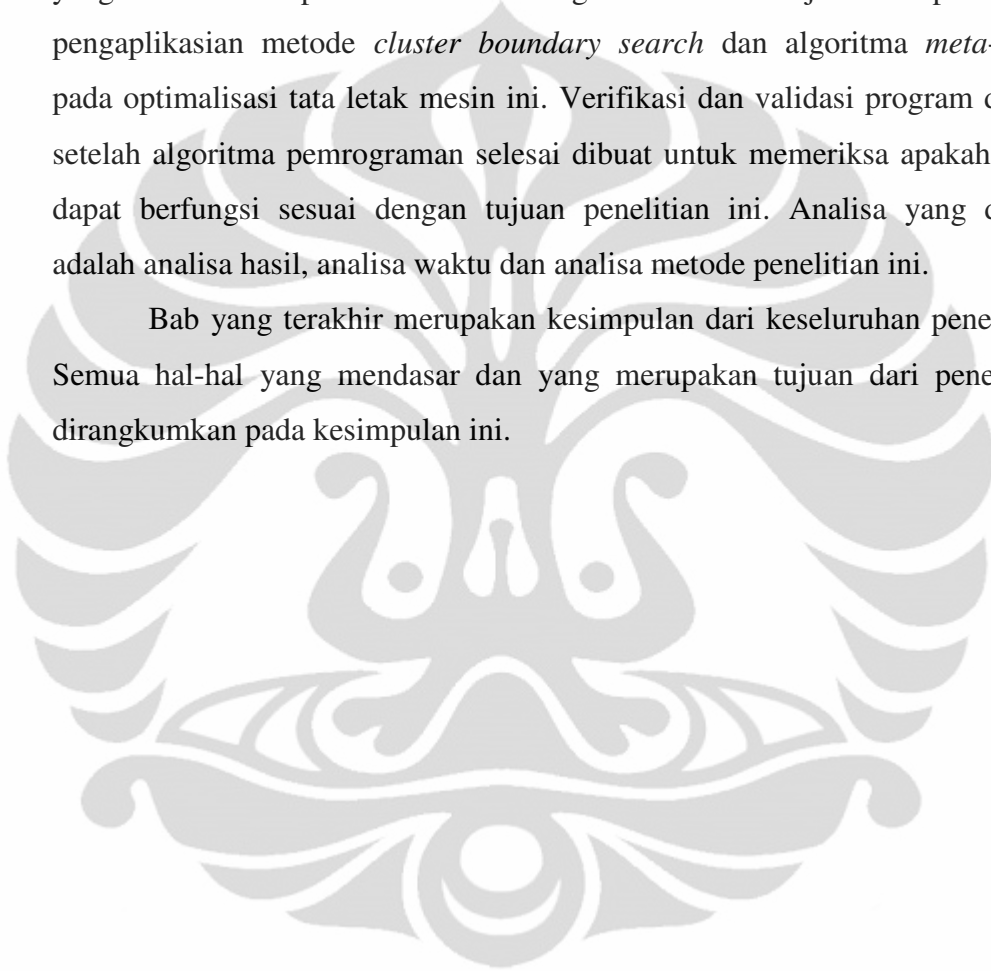


Gambar 1.2 Diagram Alir Metodologi Penelitian

Bab ketiga merupakan pengumpulan data yang dibutuhkan penulis dalam melakukan penelitian ini. Data tersebut mencakup jumlah mesin, dimensi ukuran mesin, data aliran material, karakteristik data dan sumber pengambilan data tersebut.

Bab keempat menjelaskan tentang pengolahan dan analisa data dari data yang telah dikumpulkan di bab ketiga. Peneliti menjelaskan pola berpikir pengaplikasian metode *cluster boundary search* dan algoritma *meta-heuristic* pada optimalisasi tata letak mesin ini. Verifikasi dan validasi program dilakukan setelah algoritma pemrograman selesai dibuat untuk memeriksa apakah program dapat berfungsi sesuai dengan tujuan penelitian ini. Analisa yang dilakukan adalah analisa hasil, analisa waktu dan analisa metode penelitian ini.

Bab yang terakhir merupakan kesimpulan dari keseluruhan penelitian ini. Semua hal-hal yang mendasar dan yang merupakan tujuan dari penelitian ini dirangkumkan pada kesimpulan ini.



BAB 2

LANDASAN TEORI

2.1 Machine Layout Problem

2.1.1 Definisi

Machine Layout Problem merupakan permasalahan tata letak mesin yang digambarkan dalam bentuk kotak-kotak persegi dan merupakan bagian dari *facility layout problem* yang menentukan pengaturan fisik pada sistem produksi sebuah industri. Sistem produksi ini dapat diaplikasikan ke berbagai sektor industri, termasuk sektor jasa, telekomunikasi dan lainnya.

2.1.2 Facility Layout Problem/ Permasalahan Tata Letak Fasilitas

Facility Layout Problem (FLP) didefinisikan sebagai penentuan struktur fisik dari sebuah sistem produksi (Meller, 1996). Penelitian tentang optimasi kombinatorial ini berkembang di berbagai jenis fasilitas produksi, yang mencakup industri jasa dan komunikasi. Dalam hal ini, penelitian ini akan berfokus pada *layout* fasilitas manufaktur. Proses manufaktur merupakan komponen yang penting dalam sistem produksi, karena itu *layout* yang baik itu perlu dirancang secara maksimal.

FLP berfokus pada pencarian susunan yang paling efisien dari permasalahan n departemen dengan kebutuhan area yang tidak sama pada sebuah fasilitas. Tujuan dari FLP ini tidak lain adalah untuk meminimalkan biaya *material handling* di dalam sebuah fasilitas. Beberapa tujuan lain yang ingin dicapai dari FLP adalah (Muther, 1955) :

- Memiliki aliran material yang seimbang untuk mengurangi *bottleneck* dalam produksi.
- Mengurangi jarak pemindahan material.
- Menggunakan ruang secara efektif.
- Meningkatkan kepuasan dan keamanan pekerja.
- Memperoleh fleksibilitas sehingga dapat dengan mudah diatur ulang untuk perubahan kondisi.

Batasan-batasan dalam FLP yang dijelaskan oleh Meller (1996), yaitu:

1. Kebutuhan area lantai (*floor*) dan departemen.
2. Batasan lokasi penempatan departemen, dimana departemen tidak boleh *overlap*, harus diletakan di dalam sebuah fasilitas dan dalam beberapa permasalahan tidak dapat diletakan di area tertentu.

Istilah *Dynamic Facility Layout Problem* (DFLP) digunakan ketika adanya perubahan aliran material (*material flows*) selama perencanaan tata letak. Adapun beberapa faktor yang dapat mempengaruhi perubahan aliran material adalah (Shore and Tompkins, 1980):

- Perubahan desain produk
- Penambahan atau penghapusan produk yang diproduksi
- Penggantian peralatan produksi sebelumnya
- *Life cycle* produk yang memendek
- Perubahan jumlah produksi dan jadwal produksi.

Ada beberapa jenis FLP, yaitu *Quadratic Assignment Problem* (QAP), *Unequal Area Facility Layout Problems* (UA-FLP), dan MLP. Perbandingan ketiganya dapat dilihat pada tabel 2.1

Tabel 2.1 Jenis-Jenis FLP

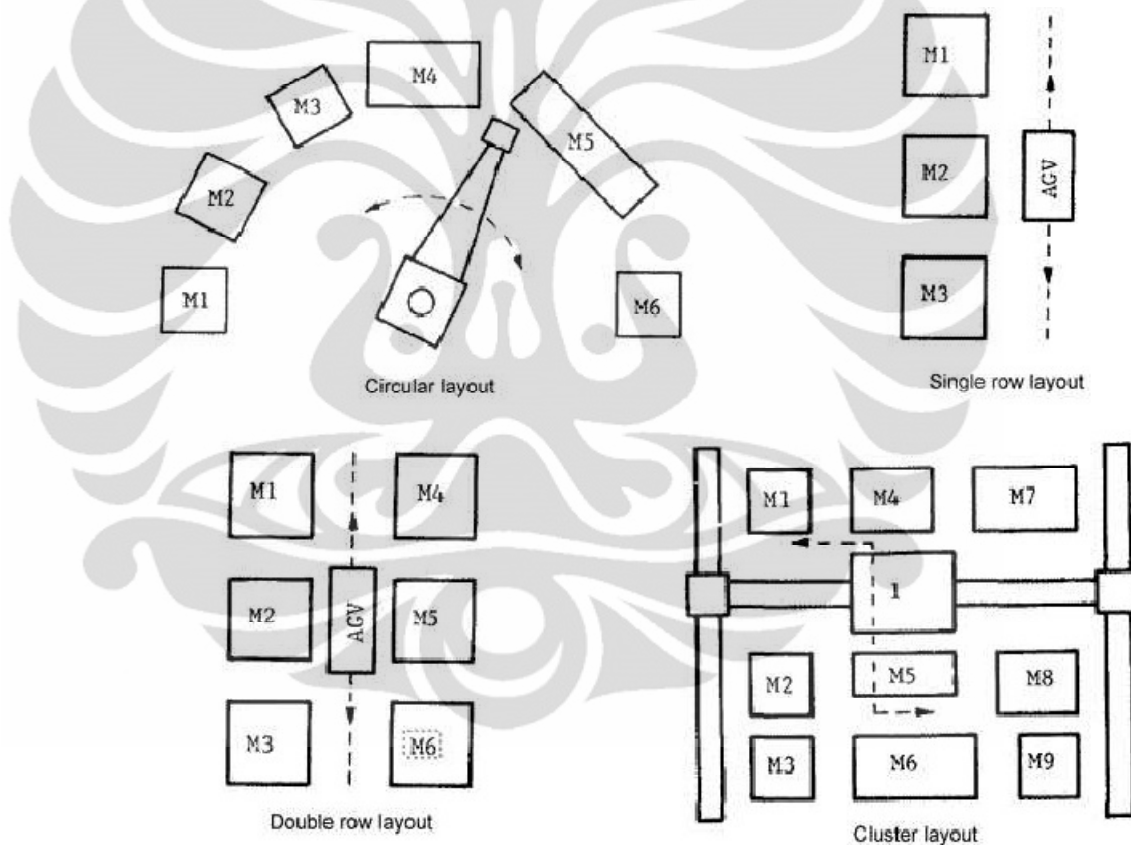
No	Permasalahan	Ukuran Departemen	Kandidat Lokasi	Area Fasilitas
1	QAP	Ukuran sama, dimensi tetap atau diabaikan	Lokasi tetap	Tetap atau diabaikan
2	UA-FLP	Ukuran berbeda, variabel keputusan	Variabel keputusan	Total area departemen
3	MLP	Ukuran berbeda, dimensi tetap	Variabel keputusan	Total area departemen atau dimensi bebas

2.1.3 Metode Penyelesaian MLP

Koopmans and Beckman (1957) pertama kali merumuskan permasalahan pengaturan fasilitas ke dalam sebuah jumlah lokasi yang sama sebagai *Quadratic Assignment Problem* (QAP). Formulasi berikutnya dari FLP berusaha untuk melinearakan QAP menjadi permasalahan program integer linear (Lawler, 1963), *mixed-integer programming problem* (Kaufman dan Broeckx, 1978; Bazaraa dan Sherali, 1980; Burkard dan Bonniger, 1983), atau permasalahan *quadratic set covering* (Bazaraa, 1975). Love dan Wong (1976a) menyajikan sebuah formulasi

programa linteger linear campuran yang bukan merupakan hasil linearisasi dari QAP. Heragu dan Kusiak (1991) juga menyajikan formulasi program integer campuran linear yang mereka selesaikan dengan teknik minimalisasi tanpa batasan (*constraint*). Hall (1970) dan Drezner (1980) merumuskan FLP sebagai program matematika *non-convex*.

Heragu dan Kusiak (1988) melakukan penelitian menggunakan *triangle assignment algorithm* untuk tata letak berbentuk *single-row*, *double-row*, dan *cluster*. Tata letak mesin secara umum biasa menggunakan pola-pola di atas. Di samping itu, pola ini sudah memiliki baris sendiri dalam meletakkan mesin yang berbeda dengan tata letak mesin secara kontinyu, seperti pada gambar 2.1.



Gambar 2.1 Tata Letak Mesin Diskrit

Algoritma *Simulated Annealing* (SA) digunakan untuk memecahkan permasalahan tata letak mesin *cellular* oleh Souliah (1995) dengan cara melakukan pengelompokan mesin ke dalam sel-sel yang sesuai dengan iterasi. Selain itu, *genetic algorithm* juga digunakan untuk memecahkan tata letak *cellular* oleh Gupta (1996). Dalam penelitian yang ia lakukan, pengaturan *cell*

hanya dilakukan pada *linear single row* dan *linear double row* dengan tidak memperhitungkan tata letak yang sebenarnya.

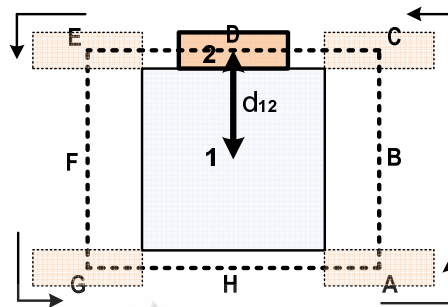
Yang dan Peters (1998) melakukan penelitian tentang permasalahan tata letak mesin untuk disesuaikan dengan layout *job shop* dengan tujuan untuk meminimalisasi biaya *material handling*. Selain itu, penelitian mereka juga berfokus pada biaya pemasangan mesin dan perancangan ulang mesin. Fleksibilitas tata letak mesin ini berhubungan erat dengan penggantian peralatan produksi dan kemampuan adaptasi dengan peralatan baru. Mereka memodelkan *machine layout problem* statis sebagai *Reduced Integer Programming* (RIP) namun tidak dapat digunakan secara riil. Penelitian ini memberikan solusi lain, yaitu *Ant Colony Optimization* (ACO) untuk memecahkan permasalahan ini.

Algoritma ACO juga digunakan oleh Corry dan Koza (2004) untuk meneliti permasalahan tata letak mesin yang bersifat *robust*. Hasil ini menunjukkan perbaikan hasil dari penelitian dengan RIP sebesar 10%-25% pada biaya relokasi dan 5%-7% pada biaya *material handling*.

Andersen (2006) membandingkan tiga metode untuk permasalahan tata letak mesin, yang mencakup RIP, ACO dan SA. Hasil penelitian tersebut menunjukkan bahwa hasil dari ACO dan SA lebih baik daripada RIP. Selanjutnya, perbandingan antara ACO dan SA menunjukkan bahwa untuk permasalahan dengan jumlah yang banyak SA lebih baik daripada ACO dan ACO lebih baik pada keadaan sebaliknya.

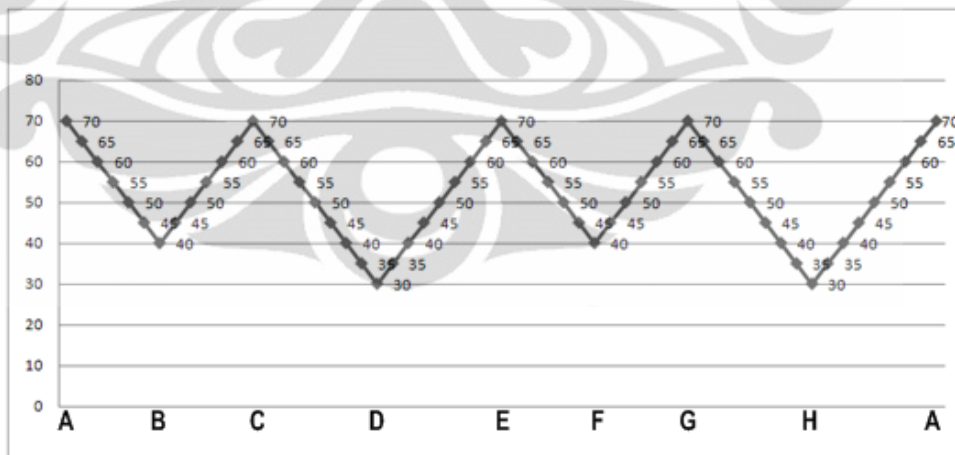
2.2 Metode *Cluster Boundary Search*

Metode *Cluster Boundary Search* (CBS) melakukan optimasi MLP dengan melakukan pencarian peletakan mesin di sekeliling batas luar mesin yang sebelumnya sudah diletakan. Pencarian keliling ini dilakukan dengan sebuah mesin pada setiap periode peletakan mesin. Semua periode peletakan mesin ini dilakukan dengan arah keliling yang sama. Pada awalnya, mesin yang pertama kali dipilih tidak melakukan proses pengelilingan ini. Proses pengelilingan ini dilakukan pada proses peletakan mesin kedua di sekeliling dari mesin pertama seperti pada gambar 2.1



Gambar 2.2 Proses Pengelilingan Mesin Pertama oleh Mesin Kedua

Proses pengelilingan mesin kedua dimulai pada titik A dan dilakukan dengan arah yang berlawanan dengan jarum jam sampai kembali ke titik awal. Dalam contoh ini, peletakan mesin diatur untuk bergerak dengan setiap '1' satuan jarak secara berurutan. Dalam proses CBS yang sesungguhnya, nilai perpindahan dari peletakan mesin ini dihasilkan dari nilai panjang dan lebar yang terkecil dari suatu MLP dan dibagi sepuluh. Nilai ini berbeda-beda karena bergantung pada jenis MLP yang sedang dioptimasi. Jarak antara batas keliling dan batas luar mesin yang sudah diletakan didapatkan dari setengah lebar dan tinggi dari mesin yang melakukan proses keliling. Ketika mesin kedua, dalam gambar 2.1, mengelilingi mesin pertama, nilai fungsi objektif dari setiap peletakan dihitung seperti pada gambar 2.2.

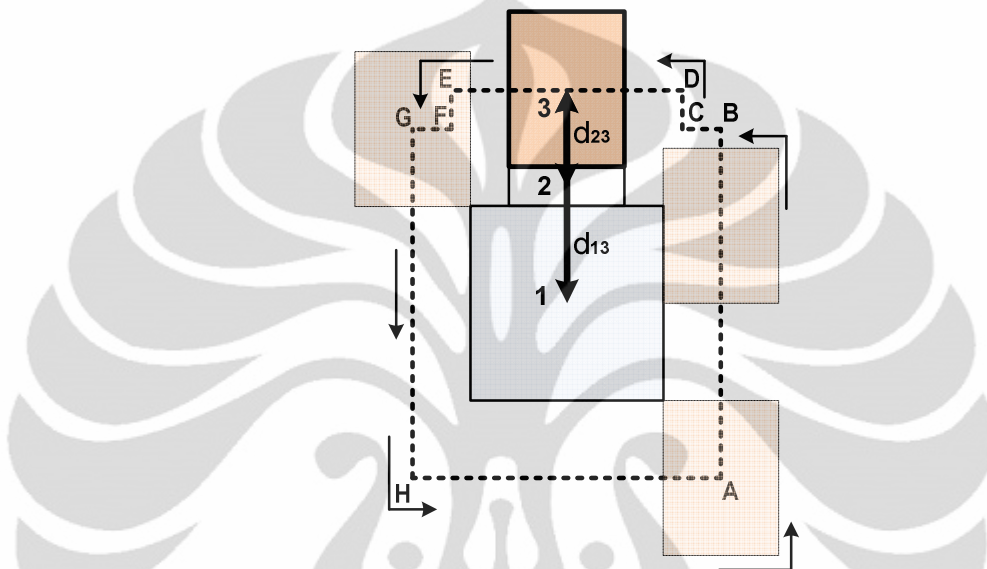


Gambar 2.3 Grafik Nilai Fungsi Objektif Peletakan Mesin Kedua dalam Proses Keliling

Grafik di atas memperlihatkan bahwa biaya *material handling* berubah-ubah pada peletakan yang berbeda-beda. Dua peletakan yang paling optimal dari mesin kedua terletak pada titik D dan H. Dalam hal ini, peletakan yang dipilih adalah titik D, karena MATLAB akan selalu memilih nilai optimal pertama yang

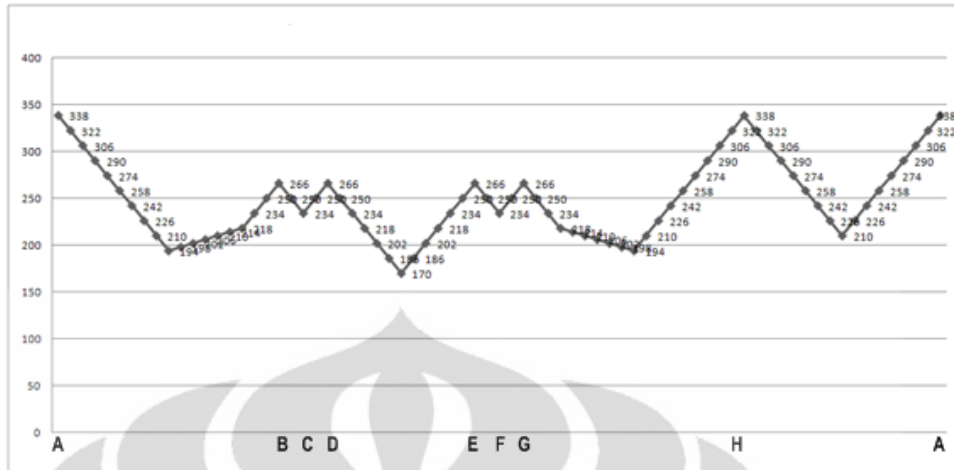
ditemui dengan nilai yang sama. Jika ada kemungkinan lain peletakan yang lebih optimal, algoritma *simulated annealing* dan *tabu search* akan melakukan pencarian layout mesin yang paling optimal.

Setelah peletakan dari mesin kedua pada titik D, batas keliling yang baru akan dihasilkan. Batas keliling baru yang terbentuk akan tergantung pada bentuk mesin pertama dan mesin kedua.



Gambar 2.4 Proses Pengelilingan Mesin Pertama dan Kedua oleh Mesin Ketiga

Proses pengelilingan yang sama dilakukan untuk meletakkan mesin ketiga, yang dimulai pada titik A. Nilai fungsi objektif pada setiap peletakan juga dihitung dan menghasilkan plot grafik seperti pada gambar 2.4. Nilai ini didapatkan dari fungsi objektif mesin ketiga terhadap mesin pertama dan mesin kedua. Fluktuasi hasil peletakan menjadi lebih kompleks dengan jumlah mesin yang lebih banyak.



Gambar 2.5 Grafik Nilai Fungsi Objektif Peletakan Mesin Ketiga dalam Proses Keliling

Gambar 2.4 menunjukkan bahwa posisi yang paling optimal untuk peletakan mesin ketiga berada diantara titik D dan E. Total biaya minimum dari ketiga mesin dengan layout ini adalah 10, dengan pengaturan biaya per unit jarak sebesar '1'.

Tabel 2.2 Perhitungan Optimasi dari Gambar 2.4

Machine		Amount	Cost per distance unit	Distance	Cost
From	To				
2	1	5	1	6	30
1	3	10	1	11	110
3	2	6	1	5	30

Dalam proses ini MATLAB juga menghasilkan koordinat posisi dari setiap mesin yang dapat dilihat pada tabel di bawah.

Tabel 2.3 Koordinat Mesin Hasil Optimasi

Machine	X Coordinate	Y Coordinate
1	0	0
2	0	5
3	0	11

Metode CBS hanya menghasilkan sebuah solusi. Walaupun proses ini dijalankan berulang kali, hasil yang didapatkan akan sama. Untuk mengantisipasi keadaan ini, solusi random digunakan untuk menemukan probabilitas solusi lain yang akan dilakukan oleh *simulated annealing* dan *tabu search*. Hal ini dilakukan dengan mengacak urutan penempatan mesin, yang menentukan prioritas pengelilingan mesin. Setelah pengacakan ini dilakukan, langkah-langkah CBS

yang sama dilakukan dan nilai fungsi objektif akan dibandingkan. Bila ada penurunan nilai, maka solusi baru akan diterima.

Sebuah fungsi objektif untuk menentukan mesin yang pertama kali diletakan atau dikelilingi, yaitu mesin yang memiliki konektivitas terbesar dengan semua mesin yang lain. Hal ini berdasarkan pada berapa banyak material yang sebuah mesin berikan kepada semua mesin lainnya dan dipindahkan ke mesin ini. Fungsi ini dituliskan sebagai berikut:

$$P_i = \text{Max} \sum_{i=1}^n F_i T_i$$

n = total jumlah mesin.

T_i = jumlah material yang dipindahkan ke mesin i.

F_i = jumlah material yang dipindahkan dari mesin i.

Fungsi ini akan dapat menentukan sebuah mesin yang pertama kali diletakan pada pusat layout untuk dikelilingi oleh mesin-mesin lainnya.

Selanjutnya, prinsip yang sama digunakan untuk menentukan secara berurutan mesin” yang akan mengelilingi mesin yang telah diletakan sebelumnya dengan konektivitas terbesar ‘Sk’.

$$S_k = \text{Max} \sum_k^{nk} \sum_i^{ni} F_{ik} T_{ki}$$

k = mesin yang belum diletakan.

nk = total jumlah mesin yang belum diletakan.

i = mesin yang sudah diletakan.

ni = total jumlah mesin yang sudah diletakan.

F_{ik} = jumlah material yang dipindahkan dari mesin i ke mesin k.

T_{ki} = jumlah material yang dipindahkan dari mesin k ke mesin i.

Fungsi ini akan dijalankan secara terus-menerus untuk memilih sebuah mesin pada setiap periode pengelilingan dan peletakan mesin. Pada akhirnya, sebuah urutan mesin akan didapatkan, dengan jumlah elemen sesuai dengan permasalahan ‘n’ mesin yang digunakan. Di dalam beberapa periode peletakan, nilai fungsi *Sk* mungkin akan memiliki nilai maksimal yang sama. Dalam hal ini MATLAB akan selalu memilih nilai maksimal yang pertama kali ditemui. Untuk mengantisipasi kemungkinan urutan mesin yang menghasilkan layout mesin yang

lebih optimal, pengacakan urutan mesin akan terus dilakukan dengan *simulated annealing* dan *tabu search* dengan parameter khusus.

2.3 Algoritma Simulated Annealing

Simulated Annealing (SA) merupakan algoritma iteratif yang menggunakan *threshold* positif dan stokastik. *Threshold* akan semakin turun seiring dengan berjalannya waktu. Karena itu, semakin mendekati akhir proses algoritma, solusi-solusi yang tidak melakukan perbaikan memiliki probabilitas yang semakin kecil untuk diterima. Perubahan solusi saat ini menjadi solusi tetangga disebut dengan langkah (*move*). *Move* yang digunakan dalam proses optimalisasi di penelitian ini adalah swap, insert, 2Opt, pergeseran ke kiri bawah dan tengah *layout*.

2.3.1 Konsep Dasar *Simulated Annealing*

Proses annealing dapat didefinisikan sebagai penurunan temperatur secara teratur atau konstan pada benda padat yang sebelumnya sudah dipanaskan sampai keadaan dimana benda tersebut mencapai *ground state/ freezing point*, atau dengan kata lain benda mencapai titik bekunya. Suhu dikurangi secara kontinu dan hati-hati sehingga pada setiap tingkatan suhu tercapai keseimbangan termal. Jika penurunan suhu tidak dilakukan secara teratur, benda padat tersebut akan memiliki kecacatan struktur karena terbentuknya struktur lokal saja yang optimal. Proses yang hanya menghasilkan struktur lokal yang optimal disebut dengan *rapid quenching*. Konsep mekanika statistik diasosiasikan dengan permasalahan optimasi dapat dilihat pada tabel 2.2 (Johnson, et al., 1989).

Tabel 2.4 Analogi Sistem Fisis dengan Proses Optimasi

Sistem Fisis	Optimasi
<i>State</i>	Solusi <i>Feasible</i>
<i>Energy</i>	Biaya
<i>Ground State</i>	Solusi Optimal
<i>Rapid Quenching</i>	Pencarian Lokal
<i>Careful Annealing</i>	<i>Simulated Annealing</i>

SA menggunakan konsep *neighbourhood search* atau *local search* dan pada setiap iterasi SA melakukan pencarian tersebut. Jika biaya *material handling*

hasil pencarian lebih baik daripada biaya dengan Layout MLP semula, maka *move* yang diambil memperbaiki (*improve*) fungsi tujuan. Semua *move* yang memperbaiki fungsi tujuan akan diambil dan disimpan, sedangkan *move* yang lebih buruk akan disimpan dengan probabilitas p' .

2.3.2 Langkah-langkah *Simulated Annealing*

Langkah-langkah SA dapat dinyatakan sebagai berikut:

1. Langkah pertama: *Initialize*
Solusi pertama kali yang dihasilkan merupakan urutan penempatan mesin berdasarkan *connectivity based ordering*. Keadaan ini dimulai pada suhu awal yang tinggi.
2. Langkah kedua: *Move*
Pergerakan dilakukan pada urutan penempatan mesin (*machine sequence*). Beberapa pergerakan yang dilakukan adalah dengan mengaplikasikan solusi *swap*, *insert*, *2Opt* dan pergeseran titik koordinat mesin.
3. Langkah ketiga: *Calculate score*
Perhitungan fungsi objektif dilakukan berdasarkan titik koordinat mesin yang dihasilkan dari pergerakan pada langkah kedua.
4. Langkah keempat: *Choose*
Perubahan nilai fungsi objektif akan menentukan apakah solusi hasil iterasi akan dipilih sebagai solusi terbaik. Jika hasil solusi iterasi lebih optimal, solusi terbaik akan menggunakan solusi iterasi tersebut dan sebaliknya. Selain itu, solusi iterasi juga disimpan sebagai solusi saat ini yang nilainya dapat berubah naik atau turun. Probabilitas diterimanya solusi iterasi sebagai solusi saat itu, jika nilainya lebih buruk daripada solusi saat itu, bergantung pada selisih nilai fungsi objektif solusi iterasi dan solusi saat itu, bilangan random dan suhu saat itu. Proses ini akan terus di-*update* seiring dengan adanya kemungkinan solusi baru yang ditemukan dengan nilai fungsi objektif yang lebih optimal.
5. Langkah kelima: *Update and repeat*

Pencatatan solusi baru dan penurunan tingkat suhu dilakukan. Selanjutnya, pencarian solusi baru kembali dilakukan (Langkah kedua). Proses ini berlangsung sampai titik suhu mencapai *freezing point* yang ditentukan.

2.4 Algoritma Tabu Search

Tabu search pertama kali diperkenalkan oleh Fred Glover pada tahun 1970. Metode ini merupakan salah satu metode pemecahan permasalahan optimasi kombinatorial yang termasuk dalam metode *local search*. Metode ini mencoba untuk meng-*improve* kekurangan pada metode klasik yang kurang cocok untuk digunakan pada permasalahan yang tergolong ke dalam *NP-hard* (Glover and Laguna, 1997). *Tabu search* sendiri telah digunakan pada berbagai bidang, diantaranya resource planning, telekomunikasi, penjadwalan, *logistic space planning* dan lainnya.

2.4.1 Pengertian Tabu Search

Tabu Search (TS) berdasar pada premis bahwa pemecahan masalah yang cerdas haruslah menggunakan memori yang adaptif (*adaptive memory*) dan eksplorasi yang responsif (*responsive exploration*). Fitur *adaptive memory* dan *responsive exploration* dalam TS membuat implementasi prosedur yang dapat melakukan pencarian himpunan solusi secara efisien dan efektif. Karena *local search* diarahkan oleh informasi yang dikumpulkan selama pencarian, TS berbeda dengan desain tanpa memori yang sangat bergantung pada proses semi acak yang menggunakan bentuk *sampling*. Contoh dari metode tanpa memori adalah *heuristic greedy*, pendekatan *annealing* dan *genetic*.

2.4.2 Langkah-langkah Tabu Search

Tabu search memiliki langkah-langkah yang digunakan dalam pencarian solusi dalam pemecahan permasalahan optimasi kombinatorial, diantaranya:

1. Langkah 1 : Menentukan solusi awal (inisialisasi)

Solusi awal biasanya dipilih dari salah satu *Basic Dispatching Rules* yang ada. Selanjutnya solusi ini akan menjadi titik awal proses pencarian solusi-solusi lain yang mendekati optimal.

2. Langkah 2 : Pembentukan solusi tetangga (*neighborhood*)

Proses pembentukan sejumlah solusi tetangga didapat melalui *move* (gerakan), yang menghasilkan sebuah solusi tetangga. *Move* yang dilakukan tidak boleh *tabu*, dalam artian gerakan yang terdapat di dalam *tabu list* tidak boleh digunakan lagi untuk mencari solusi tetangga. Himpunan solusi-solusi tetangga ini kemudian disebut juga dengan *candidate list*.

3. Langkah 3 : Pilih kandidat terbaik dari himpunan solusi tetangga

Untuk setiap *neighborhood* yang ada dalam *candidate list*, dipilih satu solusi yang mempunyai nilai fungsi tujuan paling optimum. Setiap solusi dievaluasi dengan cara membandingkan besarnya nilai fungsi objektif. Solusi yang mempunyai fungsi objektif terkecil dibandingkan dengan solusi-solusi lainnya dalam *candidate list* dan dipilih mejadi kandidat terbaik. Gerakan (*move*) yang membentuk solusi yang menjadi kandidat terbaik ini tidak berada pada *tabu list*. Jika gerakan yang akan dilakukan dilarang oleh *tabu list*, maka dipilih solusi lain yang gerakannya tidak dilarang oleh *tabu list*. Selanjutnya, gerakan yang digunakan untuk membentuk kandidat terbaik ini dimasukan ke dalam *tabu list*. Jika kandidat terbaik ini mempunyai nilai fungsi tujuan yang lebih baik dari solusi terbaik saat itu, maka kandidat terbaik ini aka menggantikan solusi terbaik yang sebelumnya dipakai.

4. Langkah 4 : Bentuk himpunan solusi tetangga baru

Kandidat terbaik yang didapat pada langkah 3 kemudian akan menjadi solusi baru yang akan dipakai untuk mencari solusi-solusi tetangga lainnya. Kemudian ulangi langkah 2 dan 3 sampai *stopping criteria* tercapai. *Stopping criteria* dapat berupa salah satu dari kondisi-kondisi berikut ini (Hertz and Taillard, 1995):

1. Solusi yang dihasilkan telah mencapai nilai tertentu yang diharapkan
2. Tidak didapatkan *neighborhood* dari solusi yang dihasilkan
3. Jumlah iterasi telah melampaui batas maksimum iterasi yang diperbolehkan
4. Jumlah iterasi melampaui batas jumlah maksimum iterasi yang ditentukan.

BAB 3

PENGUMPULAN DATA

Data utama yang dibutuhkan untuk untuk melakukan penelitian *machine layout problem* ini adalah tabel aliran material dan dimensi mesin. Dalam penelitian ini, dilakukan simplifikasi permasalahan tata letak pabrik yaitu dimana semua mesin dianggap sebagai segiempat. Perhitungan *alley*, area kerja dan kebutuhan area lainnya dapat diasumsikan tercakup maupun tidak dalam perhitungan luas area mesin ini.

Data aliran material yang peneliti gunakan dalam perhitungan ini merupakan aliran material searah (*one-side matrix*). Karena data aliran material dari satu mesin ke mesin lainnya sama jumlahnya dan sebaliknya, misalnya jumlah material dari mesin dua ke mesin empat dan sebaliknya sama, peneliti menggunakan salah satu dari aliran material tersebut. Data tabel aliran material yang diambil dapat merupakan segitiga atas atau segitiga bawah karena keduanya sama saja.

Ada beberapa permasalahan tata letak mesin yang telah dioptimasi oleh para peneliti terdahulu yang peneliti gunakan dalam penelitian ini untuk melihat perbandingan hasil optimasi yang terbaik. Daftar set data dapat dilihat pada tabel 3.1. dan untuk data keseluruhan secara detail dapat dilihat pada lampiran 1.

Tabel 3.1 Daftar Set Data

No	Problem	Jumlah Mesin	Referensi
1	KH1	4	Heragu, S.S. and Kusiak, A. (1988)
2	KH2	4	Heragu, S.S. and Kusiak, A. (1988)
3	KH3	4	Heragu, S.S. and Kusiak, A. (1988)
4	KH4	4	Heragu, S.S. and Kusiak, A. (1988)
5	KH5	4	Heragu, S.S. and Kusiak, A. (1988)
6	KH6	4	Heragu, S.S. and Kusiak, A. (1988)
7	KH7	4	Heragu, S.S. and Kusiak, A. (1988)
8	KH8	4	Heragu, S.S. and Kusiak, A. (1988)
9	KH9	4	Heragu, S.S. and Kusiak, A. (1988)
10	NUG5	5	Nugent, C.E., Vollmann, T.E. and Ruml, J. (1968)
11	NUG6	6	Nugent, C.E., Vollmann, T.E. and Ruml, J. (1968)

Tabel 3.1 Daftar Set Data (sambungan)

No	Problem	Jumlah Mesin	Referensi
12	NUG7	7	Nugent, C.E., Vollmann, T.E. and Ruml, J. (1968)
13	NUG8	8	Nugent, C.E., Vollmann, T.E. and Ruml, J. (1968)
14	NUG12	12	Nugent, C.E., Vollmann, T.E. and Ruml, J. (1968)
15	NUG15	15	Nugent, C.E., Vollmann, T.E. and Ruml, J. (1968)
16	NUG20	20	Nugent, C.E., Vollmann, T.E. and Ruml, J. (1968)
17	OPT8	8	VIP - PLANOPT Engineering Optimization Software
18	OPT12	12	VIP - PLANOPT Engineering Optimization Software
19	OPT20	20	VIP - PLANOPT Engineering Optimization Software
20	OPT28	28	VIP - PLANOPT Engineering Optimization Software
21	OPT50	50	VIP - PLANOPT Engineering Optimization Software
22	OPT75	75	VIP - PLANOPT Engineering Optimization Software
23	OPT100	100	VIP - PLANOPT Engineering Optimization Software
24	OPT125	125	VIP - PLANOPT Engineering Optimization Software
25	DUN62	62	Dunker, T. and Radons, G. (2003)

Beberapa peneliti permasalahan tata letak mesin di atas memiliki karakteristik set data yang diuji. Ada peneliti yang berfokus pada permasalahan dengan jumlah yang sedikit, banyak ataupun keduanya. Karakteristik data penelitian dapat dilihat pada tabel 3.2.

Tabel 3.2 Karakteristik Data

No	Problem	Jumlah Problem Diuji	Ukuran Minimum Problem	Ukuran Maksimum Problem
1	Heragu, S.S. and Kusiak, A. (1988)	17	4	30
2	Nugent, C.E., Vollmann, T.E. and Ruml, J. (1968)	8	5	30
3	VIP - PLANOPT Engineering Optimization Software	12	3	125
4	Dunker, T. and Radons, G. (2003)	1	62	62

Penelitian ini hanya menggunakan set data yang dipublikasikan, oleh karena itu penelitian yang penulis lakukan hanyalah menggunakan data yang terpublikasi. Daftar set data terpublikasi yang digunakan oleh peneliti terdahulu dapat dilihat pada tabel 3.3.

Tabel 3.3 Daftar Publikasi Yang Melakukan Pengolahan Data Terhadap Data Set

No	Problem	Metode					
		<i>TAA for Single Row</i>	<i>TAA for Double Row</i>	<i>TAA for Cluster MLP</i>	<i>Coevolutionary Algorithm</i>	HOT	Tabu Search/ Cluster Boundary Search
1	KH1	Heragu and Kusiak. (1988)					
2	KH2	Heragu and Kusiak. (1988)					
3	KH3	Heragu and Kusiak. (1988)					
4	KH4	Heragu and Kusiak. (1988)					
5	KH5	Heragu and Kusiak. (1988)					
6	KH6	Heragu and Kusiak. (1988)					
7	KH7	Heragu and Kusiak. (1988)					
8	KH8	Heragu and Kusiak. (1988)					
9	KH9	Heragu and Kusiak. (1988)					
10	NUG5	Heragu and Kusiak. (1988)	Heragu and Kusiak. (1988)				
11	NUG6	Heragu and Kusiak. (1988)	Heragu and Kusiak. (1988)				
12	NUG7	Heragu and Kusiak. (1988)	Heragu and Kusiak. (1988)				

Tabel 3.3 Daftar Publikasi Yang Melakukan Pengolahan Data Terhadap Data Set
(sambungan)

No	Problem	Metode					
		<i>TAA for Single Row</i>	<i>TAA for Double Row</i>	<i>TAA for Cluster MLP</i>	<i>Coevolutionary Algorithm</i>	HOT	Tabu Search/ Cluster Boundary Search
13	NUG8	Heragu and Kusiak. (1988)	Heragu and Kusiak. (1988)				
14	NUG12	Heragu and Kusiak. (1988)	Heragu and Kusiak. (1988)	Heragu and Kusiak. (1988)			
15	NUG15	Heragu and Kusiak. (1988)	Heragu and Kusiak. (1988)	Heragu and Kusiak. (1988)			
16	NUG20	Heragu and Kusiak. (1988)	Heragu and Kusiak. (1988)	Heragu and Kusiak. (1988)			
17	OPT20						Mckendall et al. (2009)
18	OPT50					Mir and Imam (2001)	Mckendall et al. (2009)
19	OPT100						Mckendall et al. (2009)
20	OPT125						Mckendall et al. (2009)
21	DUN62				Dunker, et al. (2003)		Mckendall et al. (2009)

Selain peneliti di tabel 3.3, ada juga set data yang diambil dari *software* optimasi PLANOPT dengan label OPT. Semua hasil optimasi oleh para peneliti terdahulu maupun *software* PLANOPT dirangkum dalam tabel 3.4.

Tabel 3.4 Solusi Terpublikasi Penelitian Sebelumnya

No	Problem	Metode							
		TAA Single Row	TAA Double Row	TAA Cluster	Coevolutionary Algorithm	PLANOPT	HOT	DE	TS/BSH
1	KH1	225.00						168.00	
2	KH2	440.00						335.00	
3	KH3	510.00						360.00	
4	KH4	465.00						335.00	
5	KH5	19.68						14.60	
6	KH6	359.00						267.00	
7	KH7	318.00						248.00	
8	KH8	60.00						42.00	
9	KH9	244.00						192.00	
10	NUG5	1.17	1.14					0.77	
11	NUG6	2.09	2.01					1.36	
12	NUG7	5.42	3.98					2.64	
13	NUG8	8.00	4.95					4.37	
14	NUG12	31.53	17.91	15.77				13.09	
15	NUG15	62.62	34.98	29.09				21.09	
16	NUG20	178.15	91.47	70.86				51.33	
17	OPT8					213.50		192.50	
18	OPT12					5384.43		2918.98	
19	OPT20					1157.00		1325.88	1151.40
20	OPT28					6447.25			
21	OPT50					78224.68	80794.24		71291.40
22	OPT75					34396.38			
23	OPT100					538193.10			496820.40
24	OPT125					1084451.00			1008839.00
25	DUN62				4221911.00	3939362.00		5195260.00	3812825.00

BAB 4

PENGOLAHAN DATA DAN ANALISA

4.1 Pengolahan Data

Pembuatan program optimasi ini dibuat berdasarkan data set yang telah dikumpulkan sebelumnya. *Output* program ini adalah *layout* mesin dengan biaya *material handling* terendah dan koordinat X dan Y dari titik pusat (*centroid*) setiap mesin. Variabel dinamis yang menentukan nilai fungsi objektif akhir adalah akumulasi jarak antar titik pusat setiap mesin. Variabel tetap yang tidak berubah nilainya adalah jumlah perpindahan material dan biaya material per unit jarak.

4.1.1 Perancangan Model dan Algoritma Optimasi

Pembuatan model optimasi MLP ini menggunakan MATLAB R2009b dengan dibagi menjadi beberapa tahapan sesuai dengan metode yang dipakai yaitu, *Cluster Boundary Search* (CBS), algoritma *Simulated Annealing* (SA) dan *Tabu Search* (TS). Langkah awal yang dilakukan adalah pembuatan model CBS yang menghasilkan solusi tunggal. Langkah berikutnya adalah optimasi hasil CBS dengan menggunakan SA dan TS yang menghasilkan beragam solusi lain. SA dan TS akan memilih solusi lainnya dengan biaya *material handling* yang terkecil. Dengan demikian, pengolahan data ini akan menghasilkan dua *output*, yaitu optimasi CBS dengan algoritma SA dan optimasi CBS dengan algoritma TS.

4.1.1.1 Algoritma Fungsi Objektif

Syntax ini bertujuan untuk menghasilkan biaya *material handling* dari berbagai macam permasalahan tata letak mesin yang di-*input* ke dalam program ini. Fungsi objektif biaya *material handling* dituliskan sebagai berikut.

$$C_{ij} = \text{Min} \sum_{i=1}^n \sum_{j=1}^n u_{ij} f_{ij} d_{ij}$$

n = jumlah mesin.

C_{ij} = biaya *material handling*.

u_{ij} = biaya per unit jarak antara mesin i and j.

f_{ij} = jumlah aliran material antara mesin i and j.

d_{ij} = jarak antara mesin i and j.

Perhitungan jarak antar titik tengah mesin yang digunakan menggunakan jarak *rectilinear*, *euclidian* dan *squared euclidian*.

- Jarak *Rectilinear*

$$d_{ij} = |x_i - x_j| + |y_i - y_j|$$

- Jarak *Euclidean*

$$d_{ij} = [(x_i - x_j)^2 + (y_i - y_j)^2]^{1/2}$$

- Jarak *Squared Euclidean*

$$d_{ij} = (x_i - x_j)^2 + (y_i - y_j)^2$$

Perhitungan jarak untuk MLP OPT50 dan OPT 28 menggunakan jarak *euclidian* dan *squared euclidean* secara berurutan. Untuk MLP lainnya semua menggunakan jarak *rectilinear*. Hal ini disebabkan, penelitian ini merujuk pada perhitungan jarak yang telah dilakukan sebelumnya oleh PLANOPT. Sesuai dengan penelitian sebelumnya, dalam penelitian ini biaya material per unit jarak bernilai 1. Karena itu, fungsi objektif hanya dihasilkan dari 3 segmen:

- Koordinat X setiap mesin
- Koordinat Y setiap mesin
- Matriks aliran material

Untuk mempermudah perhitungan fungsi objektif, jarak antar mesin dibentuk menjadi sebuah matriks, yang disebut sebagai matriks jarak. Pembuatan matriks ini menggunakan fungsi yang umum dipakai dalam pemrograman, yaitu *looping* untuk mengisi setiap elemen baris dan kolom dalam matriks.

```

for i = 1 : NumberOfMachines
    for j = 1 : NumberOfMachines
        MatrixDistanceMachine(i, j) = Hitungdistance(X_coordinate(i),
            Y_coordinate(i), X_coordinate(j), Y_coordinate(j));
    end
end
end

```

Setiap elemen matriks jarak dikalikan dengan setiap elemen matriks aliran material secara langsung. Selanjutnya, semua segmen dalam matriks hasil perkalian tersebut dijumlahkan. Hasil penjumlahan semua segmen matriks perkalian tersebut merupakan nilai fungsi objektif dari koordinat mesin yang di-*input*.

4.1.1.2 Pengurutan Penempatan Mesin dengan Konektivitas Terbesar

Setiap mesin memiliki urutan prioritas untuk diambil terlebih dahulu dan mengelilingi mesin yang telah diletakan sebelumnya. Prioritas ini didasarkan pada konektivitas terbesar sebuah mesin dengan setiap mesin lainnya. Mesin terpilih pertama yang dipilih untuk dikelilingi ditentukan berdasarkan nilai P_i terbesar, yaitu penjumlahan setiap baris matriks aliran material (jumlah material yang di-supply dari mesin n) dan penjumlahan setiap kolom matriks aliran material (jumlah materil yang di-supply ke mesin n). Nomor mesin terpilih yang terpilih diidentifikasi berdasarkan kolom atau baris ke- n matriks yang terpilih. Nomor mesin pertama yang terpilih dinamakan dengan 'NomorMesin1'.

Untuk mengidentifikasi mesin yang sudah dan belum terpilih, ada sebuah urutan angka nol, yang diberi nama 'SudahDigunakan', dengan jumlah yang sama dengan jumlah mesin MLP yang digunakan. Jika mesin n sudah dipilih untuk dikelilingi, urutan ke- n dari 'SudahDigunakan' akan berubah menjadi angka 1.

Fungsi *looping* dan *if* digunakan dalam melakukan proses ini sampai terbentuknya sebuah deretan prioritas peletakan mesin (*machine sequence*). Fungsi *looping* mulai dioperasikan pada pemilihan mesin berikutnya yang akan mengelilingi mesin terpilih sebelumnya. Pengulangan otomatis ini dilakukan pada mesin yang sudah dan belum terpilih. Untuk mengenali apakah mesin sudah terpilih, fungsi *if* dioperasikan pada deret 'SudahDigunakan'.

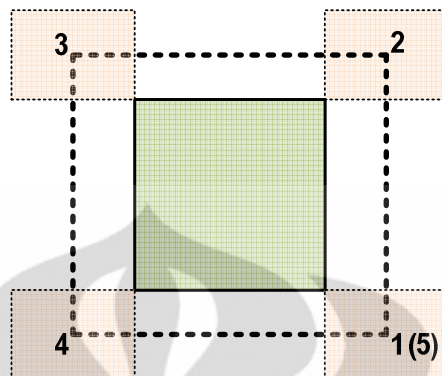
$$Sk = \text{Max} \sum_k^{nk} \sum_i^{ni} F_{ik} T_{ki}$$

Perhitungan Sk dilakukan antara mesin yang memiliki nilai 'SudahDigunakan' = 0, yaitu mesin yang akan dipilih untuk melakukan proses keliling dan mesin yang memiliki nilai 'SudahDigunakan' = 1, yaitu mesin yang sudah diletakan untuk dikelilingi.

4.1.1.3 Perancangan Model Koordinat Output

Identifikasi batas-batas pengelilingan oleh mesin terpilih kedua dan selanjutnya diwujudkan dengan menentukan empat titik sudut batas pengelilingan pada awalnya. Mesin yang pertama kali diletakan berada di titik koordinat (0,0) Titik-titik sudut pengelilingan ini dibagi menjadi dua jenis, yaitu koordinat X dan

koordinat Y *centroid* mesin. Selanjutnya, Jarak batas luar dihasilkan berdasarkan setengah lebar dan tinggi mesin yang mengelilingi.

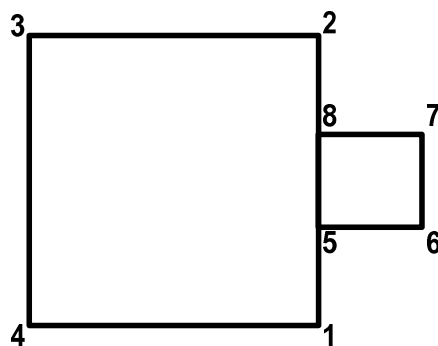


Gambar 4.1 Fase Awal Pembentukan *Boundary*

Bagian-bagian pengelilingan mesin dibagi menjadi empat bagian untuk memudahkan pembentukan *boundary* di semua sisi, yaitu:

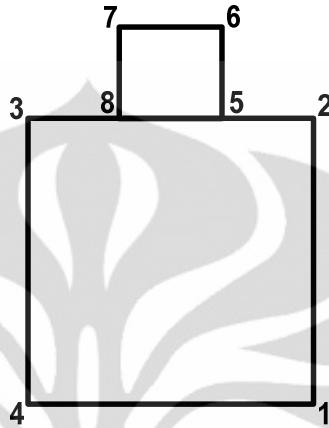
- Bagian vertikal sebelah kanan
- Bagian vertikal sebelah kiri
- Bagian horizontal sebelah atas
- Bagian horizontal sebelah bawah

Hal ini akan digunakan untuk melakukan *updating* pada *boundary* baru yang akan dibentuk selanjutnya setelah mesin sudah diletakan di posisi yang paling optimal. Urutan pembacaan yang dilakukan MATLAB pada gambar 4.1 berdasarkan urutan dari angka terkecil, sehingga terbaca sebagai '1 2 3 4'. Jika peletakan mesin yang melakukan pengelilingan mempunyai nilai fungsi objektif paling optimal pada peletakan seperti di gambar 4.2, maka akan ada *insertion* koordinat mesin kedua pada deret '1 2 3 4'. *Updating* urutan koordinat keseluruhan yang terjadi pada kondisi ini adalah '1 [5 6 7 8] 2 3 4'.



Gambar 4.2 Fase I *Updating Boundary*

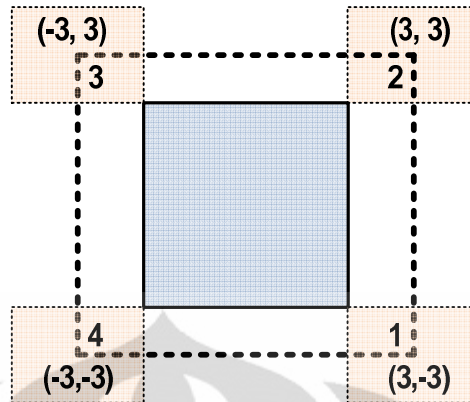
Jika peletakan mesin dengan nilai fungsi objektif optimal seperti pada gambar 4.3, maka *updating* urutan koordinat keseluruhan mesin akan berubah menjadi '1 2 [5 6 7 8] 3 4'. Proses *updating* urutan koordinat X dan Y ini terus dilakukan selama proses optimasi *cluster boundary search* dijalankan.



Gambar 4.3 Fase II Updating Boundary

Angka-angka pada deret yang terbentuk dan di-*update* terus-menerus ini selanjutnya akan menghasilkan deretan koordinat X dan Y yang akan menjadi batas pengelilingan mesin selanjutnya, baik secara vertikal maupun horizontal. Jika misalnya, titik 1 dan titik 2 pada gambar 4.3 mempunyai titik koordinat (3, -3) dan (3, 3) yang terpisah secara vertikal, maka kedua titik ini akan menjadi dasar untuk membentuk batas pengelilingan di sisi kanan vertikal dengan nilai (3, -3), (3, -2), (3, -1), (3, 0), (3, 1), (3, 2) dan (3, 3), jika nilai interval pergerakan mesin adalah 1 unit jarak. Hal yang sama juga terjadi pada titik yang terpisah secara horizontal.

Untuk mengidentifikasi posisi dari mesin yang diletakan pada empat kemungkinan posisi di batas keliling mesin sebelumnya, ada perumusan yang dibuat agar MATLAB dapat secara otomatis mengidentifikasi dan melakukan *updating* dengan melakukan *insertion* empat titik *boundary* baru pada rongga yang tepat di deret koordinat sebelumnya. Pertama-tama pembedaan yang dilakukan adalah untuk mengidentifikasi *boundary* vertikal atau horizontal. Jika $x_{(i)} = x_{(i+1)}$, maka *boundary* tersebut adalah horizontal dan jika $y_{(i)} = y_{(i+1)}$ maka *boundary* tersebut adalah vertikal (dengan i adalah deretan angka titik sudut koordinat, 1, 2, 3, 4).



Gambar 4.4 Identifikasi *Boundary* Berdasarkan Koordinat

Pada *boundary* vertikal, identifikasi yang lebih spesifik untuk mengetahui posisi mesin di kiri dan kanan *boundary* dibutuhkan.

- Jika $y_{(i)} > y_{(i+1)}$, maka MATLAB akan membentuk koordinat *boundary* sebelah kanan.
- Jika $y_{(i)} < y_{(i+1)}$, maka MATLAB akan membentuk koordinat *boundary* sebelah kiri.

Selanjutnya, pada *boundary* horizontal, MATLAB akan membentuk koordinat *boundary* yang spesifik dengan perumusan sebagai berikut:

- $x_{(i)} < x_{(i+1)}$ akan membentuk koordinat *boundary* sebelah bawah
- $x_{(i)} > x_{(i+1)}$ akan membentuk koordinat *boundary* sebelah atas

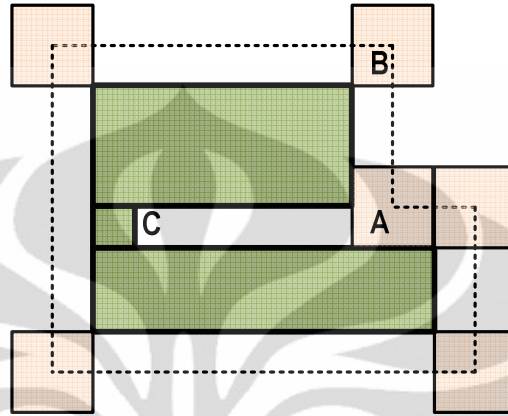
Sebelum melakukan perhitungan nilai fungsi objektif untuk menentukan posisi optimal, *constraints* diperlukan untuk mencegah mesin agar tidak menumpuk (*overlapping*). Ada pemberian nilai 1 jika mesin tidak bertumpuk dan nilai 0 jika mesin bertumpuk. Selanjutnya fungsi *if* digunakan untuk menentukan apakah perhitungan fungsi objektif akan dilakukan. Jika nilai 1, maka perhitungan nilai fungsi objektif akan dilakukan dan nilai 0 sebaliknya.

Perumusan *syntax* untuk mencegah overlapping yang dilakukan pada koordinat X dan Y diuraikan sebagai berikut:

- Feasible = 0 untuk $|x_i - x_j| < (wx_i + wx_j)/2$ dan $|y_i - y_j| < (wy_i + wy_j)/2$
- Feasible = 1 untuk $|x_i - x_j| > (wx_i + wx_j)/2$ dan $|y_i - y_j| > (wy_i + wy_j)/2$

dimana,

- $x_{i,j}$ = koordinat X mesin i dan j
 $y_{i,j}$ = koordinat Y mesin i dan j
 $w_{x_{i,j}}$ = lebar mesin i dan j
 $w_{y_{i,j}}$ = tinggi mesin i dan j



Gambar 4.5 Boundary Pada Keadaan Tertentu

Selanjutnya, perhitungan biaya nilai fungsi objektif dilakukan dengan fungsi tujuan C_{ij} .

$$C_{ij} = \text{Min} \sum_{i=1}^n \sum_{j=1}^n u_{ij} f_{ij} d_{ij}$$

Posisi optimal yang dipilih MATLAB adalah koordinat dengan nilai fungsi objektif terkecil yang pertama kali ditemui, jika ada beberapa nilai optimal yang sama. MATLAB juga akan mengambil nilai i yang memiliki koordinat X dan Y terpilih, untuk selanjutnya digunakan dalam *insertion* deret koordinat dan *updating boundary* mesin yang mau akan dikelilingi.

Nilai langkah interval keliling yang digunakan didapatkan dari nilai minimal lebar dan tinggi mesin MLP yang digunakan dibagi dengan angka 10. Hal ini menyebabkan nilai ini berbeda-beda, tergantung pada MLP yang sedang dioptimasi.

Secara keseluruhan, *output* koordinat X dan Y yang dihasilkan pada proses ini merupakan hasil dari *input* data:

- Urutan peletakan mesin (*machine sequence*)
- Lebar dan tinggi mesin
- Matriks aliran material

4.1.1.4 Improvisasi *Layout*

Beberapa improvisasi dilakukan untuk mendapatkan nilai fungsi objektif yang lebih baik. Ide ini diadaptasikan dari jurnal '*Paper Comparison of techniques for dealing with empty spaces*' oleh Kuan Yew Wong dan Komarudin. Pencarian layout permasalahan mesin terbaik dilakukan dengan menggeser koordinat seluruh mesin ke sisi tengah (*centre of the facility*) dan kiri bawah (*left bottom of the facility*). Kedua hal ini dilakukan terpisah dan digabungkan pada algoritma *simulated annealing* secara berurutan. Output koordinat X dan Y yang diambil dipilih berdasarkan nilai fungsi objektif terkecil yang dihasilkan.

4.1.1.5 Optimasi dengan *Simulated Annealing*

Proses ini bertujuan untuk mencari global optimum dengan beberapa parameter khusus, yaitu suhu awal, suhu akhir dan tingkat penurunan suhu.

Langkah-langkah SA dalam penelitian ini dapat dijabarkan sebagai berikut:

1. Langkah pertama: *Initialize*

Solusi pertama kali yang dihasilkan merupakan urutan penempatan mesin berdasarkan *connectivity based ordering*. Keadaan ini merupakan inisialisasi suhu awal yang tinggi. Di samping itu tingkat penurunan suhu dan nilai suhu akhir juga ditetapkan. Nilai tingkat penurunan suhu yang semakin mendekati angka satu, akan membuat waktu iterasi *simulated annealing* lebih lama.

2. Langkah kedua: *Move*

Pergerakan ini dilakukan untuk mencari *global optimum* dan dalam penelitian ini ada tiga jenis pergerakan yang dirancang:

1. Pergerakan yang pertama dilakukan pada urutan penempatan mesin (*machine sequence*). Beberapa pergerakan yang dilakukan adalah dengan mengaplikasikan solusi *swap*, *insert* dan *2Opt* secara random. Kemudian, output koordinat X dan Y didapatkan dari *machine sequence* yang baru ini akan menghasilkan nilai *objective function* yang baru pula. Nilai ini disimpan untuk kemudian dibandingkan.
2. Pergerakan yang kedua adalah dengan melakukan pergeseran koordinat X dan Y mesin yang dihasilkan dari pergerakan pertama ke

kiri bawah (*left bottom*) layout. Nilai *objective function* dari koordinat yang baru dihitung untuk melihat apakah nilai yang didapatkan lebih kecil. Jika nilai tersebut lebih baik, koordinat X dan Y dari pergerakan ke *left bottom* disimpan dan sebaliknya.

3. Pergerakan yang ketiga adalah dengan melakukan pergeseran koordinat X dan Y mesin yang dihasilkan dari langkah kedua ke tengah (*centre*) layout. Nilai *objective function* kembali dihitung untuk mengetahui apakah lebih minimal. Jika lebih optimal, nilai ini beserta koordinat X dan Y disimpan sebagai solusi terbaik.

3. Langkah ketiga: *Calculate Score*

Perhitungan fungsi objektif yang terbaik dilakukan dan disimpan, yaitu berdasarkan solusi koordinat X dan Y yang dihasilkan dari pergerakan-pergerakan di langkah kedua.

4. Langkah keempat: *Choose*

Perubahan nilai fungsi objektif akan menentukan apakah solusi tersebut akan dipilih sebagai solusi terbaik. Jika nilai tersebut lebih minimum, maka solusi tersebut akan disimpan sebagai solusi saat ini dan solusi terbaik.

Di samping itu, *simulated annealing* memiliki probabilitas untuk menerima solusi iterasi yang memiliki nilai fungsi objektif yang lebih buruk dari pada nilai solusi saat ini. Solusi ini akan disimpan dan probabilitas diterimanya solusi ini berdasarkan persamaan berikut:

$$p' = e^{-\Delta E / T}$$

ΔE = nilai solusi iterasi (hasil *move*) – nilai solusi saat ini

T = suhu sekarang

e = bilangan natural

Sebuah bilangan random (*r*) dihasilkan pada range nol sampai dengan satu dan jika $r < P$, maka solusi iterasi yang lebih buruk tersebut akan disimpan sebagai solusi saat ini. Jumlah solusi saat ini yang disimpan hanya berjumlah satu. Oleh karena itu, nilai solusi saat ini dapat berubah-ubah, naik atau turun, tergantung pada nilai *r*, selisih nilai solusi iterasi dan

solusi saat ini dan T. Sebaliknya, nilai solusi terbaik hanya akan berubah jika ada nilai hasil solusi iterasi yang lebih optimal.

5. Langkah kelima: *Update and repeat*

Pada langkah ini, hasil solusi terbaik serta solusi saat ini dicatat dan disimpan dan penurunan tingkat suhu dilakukan. Selanjutnya, SA kembali mencari solusi baru dengan langkah yang sama, yaitu melakukan *move* pada *machine sequence* (langkah kedua) dan selanjutnya. Proses ini terus berlangsung sampai T mencapai suhu akhir (*freezing point*) yang telah ditentukan.

4.1.1.6 Optimasi dengan *Tabu Search*

Metode meta-heuristic lain, *Tabu Search* (TS), juga digunakan untuk mencari global optimum pada MLP di penelitian ini. Beberapa parameter yang digunakan adalah jumlah iterasi maksimum dan jumlah solusi tetangga. Beberapa langkah aplikasi TS dalam penelitian ini adalah sebagai berikut:

1. Langkah 1 : Menentukan solusi awal (inisialisasi jadwal)

Machine sequence (urutan penempatan mesin) awal dihasilkan berdasarkan solusi random *randperm* pada MATLAB. Kemudian dengan *input* data *machine sequence*, lebar mesin, panjang mesin dan matriks aliran material, nilai fungsi objektif awal dapat dihitung. Untuk memaksimalkan nilai solusi awal, dilakukan pergeseran koordinat mesin awal ini ke kiri bawah (*left-bottom*) dan tengah (*centre*) dari layout. Layout yang menghasikan nilai fungsi objektif yang terkecil disimpan sebagai solusi awal.

Pada tahap ini, hasil solusi terbaik dan solusi saat ini hasil optimasi dinyatakan sebagai hasil solusi awal, yang mencakup *machine sequence*, koordinat X, koordinat Y dan *objective function*. Nilai *tabu list* pertama juga mengambil nilai solusi awal, yang mencakup koordinat X dan koordinat Y, untuk disimpan.

2. Langkah 2 : Pembentukan solusi tetangga (*neighborhood*)

Solusi tetangga yang dicatat dan disimpan mencakup *machine sequence*, koordinat X, koordinat Y mesin dan nilai *objective function*,

dengan jumlah maksimum sesuai parameter yang ditetapkan di awal iterasi. Solusi tetangga dihasilkan dengan cara melakukan pengacakan *machine sequence* dengan solusi *insert*, *swap* dan *2opt*. Kemudian cara yang sama seperti langkah 1 dilakukan, yaitu mengambil sebuah solusi tetangga dengan nilai fungsi objektif terbaik setelah dilakukannya pergeseran koordinat X dan Y mesin ke *left-bottom* dan *centre* dari *layout*. Proses ini dilakukan satu per satu sampai tabel solusi tetangga terisi penuh.

Hasil koordinat X dan koordinat Y mesin solusi tetangga yang dihasilkan ini kemudian dibandingkan dengan solusi koordinat X dan Y mesin pada *tabu list*. Jika solusi yang didapat sama, maka solusi tersebut dinyatakan tabu. Solusi yang dinyatakan tabu dinyatakan sebagai angka 1 dan solusi yang tidak tabu dinyatakan sebagai angka 0.

3. Langkah 3 : Pilih kandidat terbaik dari himpunan solusi tetangga

Pada langkah ini MATLAB akan memilih sebuah solusi tetangga terbaik dari kumpulan solusi tetangga yang tidak tabu dengan nilai *objective function* yang lebih kecil dari pada solusi tetangga terbaik pada saat itu. Indeks solusi tetangga tidak tabu yang terpilih disimpan untuk dibandingkan selanjutnya. Kemudian, TS memilih solusi terbaik dengan cara membandingkan nilai *objective function* dari solusi terbaik pada saat itu dengan solusi tetangga terbaik. Solusi terbaik yang disimpan mencakup *machine sequence*, koordinat X dan Y mesin, serta nilai *objective function*.

Updating solusi saat ini juga dilakukan dengan cara memasukan data dari solusi tetangga tidak tabu yang terbaik ke *tabu list*. Solusi di dalam *tabu list* akan berfungsi sebagai memori sementara TS dalam menentukan apakah pencarian solusi tetangga yang baru berikutnya tabu atau tidak. Besar kecilnya memori pada perhitungan ini ditentukan oleh penentuan panjang tabu di awal iterasi. Semakin panjang tabu, maka memori yang disimpan akan semakin banyak, namun akan membuat perhitungan berjalan lebih lama.

4. Langkah 4 : Bentuk himpunan solusi tetangga baru

Solusi tetangga baru kembali dibuat dengan kembali ke langkah kedua. Ketika solusi tetangga baru dihasilkan, MATLAB akan

menentukan solusi mana yang tabu dan tidak dengan membandingkannya dengan solusi yang disimpan di *tabu list*. Proses ini akan selesai ketika sampai jumlah iterasi maksimum tercapai atau limit waktu mencapai batas maksimum yang ditentukan.

4.1.2 Verifikasi dan Validasi Program

Tahap verifikasi merupakan proses untuk melihat kesesuaian antara model program yang telah dirancang dengan perencanaan model yang diinginkan. Parameter model program dapat dikatakan telah diverifikasi bila telah berjalan sesuai dengan konsep model perencanaan. Beberapa tahapan proses verifikasi yang dilakukan dalam proses pembuatan program optimasi ini adalah sebagai berikut:

- Memastikan kebenaran logika pemikiran, yaitu kesesuaian penulisan *syntax* dengan konsep metode *cluster boundary search*, *simulated annealing* dan *tabu search* untuk dapat dijalankan dengan baik
- Memastikan jika program telah menghasilkan *output* sesuai *constraint*
- Memastikan bahwa nilai fungsi tujuan dihasilkan dari semua elemen yang ditentukan
- Memastikan adanya perubahan *output* (uji sensitivitas) bila parameter diubah

Selanjutnya, proses validasi dilakukan untuk memastikan jika nilai output yang dihasilkan program benar sesuai dengan fungsi objektif. Dalam hal ini, uji validasi dilakukan dengan membandingkan hasil perhitungan manual dengan perhitungan oleh program. Selain itu, layout yang dihasilkan oleh program juga dibandingkan dengan perhitungan manual. Adapun set data yang digunakan untuk validasi adalah data KH4.

Tabel 4.1 Ukuran Mesin KH4 untuk Validasi Program

Nomor Mesin	Panjang	Lebar
1	2	2
2	4	4
3	6	6
4	2	2

Tabel 4.2 Tabel Aliran Material KH4 Validasi Program

	1	2	3	4
1	0	10	15	20
2	0	0	10	15
3	0	0	0	10
4	0	0	0	0

Adapun parameter yang digunakan untuk validasi program dapat dilihat pada tabel 4.3.

Tabel 4.3 Parameter Validasi Program

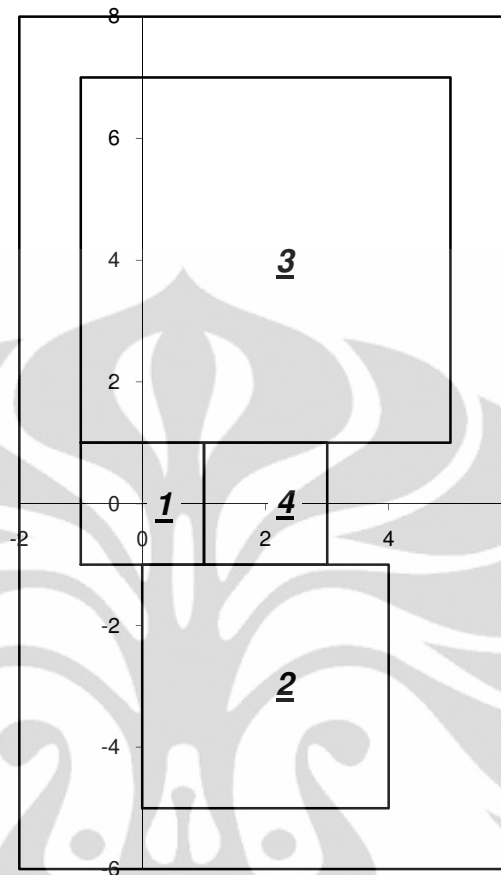
Suhu Awal	100
Suhu Akhir	10
Tingkat Penurunan Suhu	0.9999

Program MATLAB yang dijalankan menghasilkan koordinat X dan Y yang dapat dilihat pada tabel 4.4, dengan nilai fungsi objektif optimal sebesar 335.

Tabel 4.4 Output Koordinat Mesin Validasi Program

Mesin	Koordinat X	Koordinat Y
1	0	0
2	2	-3
3	2	4
4	2	0

Cluster boundary Search KH4

**Gambar 4.6** Hasil *Layout* MATLAB untuk Validasi Program

Selanjutnya, perhitungan manual dilakukan untuk membandingkan kebenaran *output* yang dihasilkan oleh program, yang mencakup nilai fungsi objektif dan *layout* mesin. Beberapa langkah yang dilakukan adalah sebagai berikut:

1. Melakukan *cluster boundary search* secara manual untuk mendapatkan hasil koordinat X dan Y *layout* mesin.

Tabel 4.5 Output Koordinat Mesin secara Manual

Mesin	X	Y
1	0	0
2	2	-3
3	2	4
4	2	0

2. Melakukan perhitungan jarak titik pusat (*centroid*) mesin secara manual yang dirangkumkan dalam sebuah matriks jarak.

Tabel 4.6 Matriks Jarak *Centroid* Mesin untuk Validasi Program

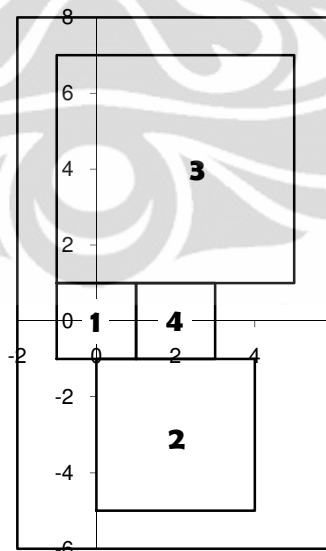
Dari/Ke	1	2	3	4
1	0	5	6	2
2	5	0	7	3
3	6	7	0	4
4	2	3	4	0

3. Melakukan perhitungan biaya *material handling*, yaitu dengan mengalikan setiap elemen di matriks jarak dengan setiap elemen di matriks perpindahan material.

Tabel 4.7 Matriks Biaya *Material Handling* untuk Validasi Program

Dari/Ke	1	2	3	4
1	0	50	90	40
2	0	0	70	45
3	0	0	0	40
4	0	0	0	0

Total hasil nilai fungsi objektif yang didapat dari perhitungan manual adalah 335. Layout yang dihasilkan juga sama dengan hasil yang dijalankan oleh program. Hal ini menunjukkan bahwa hasil yang didapatkan dari program dan perhitungan manual sama dan valid.

**Gambar 4.7** Hasil *Layout* Perhitungan Manual

Adapun untuk memunculkan layout dari MLP, *Microsoft Excel* digunakan dengan cara memasukan koordinat X dan Y yang didapatkan sebelumnya baik cari program MATLAB maupun secara manual.

4.2 Pengolahan Data dan Hasil

Di tahap ini, program optimasi MLP telah siap untuk dioperasikan untuk mengoptimasi 25 data set yang telah dikumpulkan. Pengolahan data ini dilakukan pada Laboratorium SEMS departemen Teknik Industri Universitas Indonesia, dengan spesifikasi komputer PC HP Workstation Core 2 Duo (2.3 GHz) dan *Windows operating system*. Hasil pengolahan data yang ditampilkan dibagi menjadi dua, yaitu dengan algoritma *simulated annealing* dan *tabu search*. Karena itu, parameter yang diterapkan juga dibagi menjadi dua, yaitu:

- *Simulated Annealing* (SA)

Tabel 4.8 Parameter Pengolahan Data dengan *Simulated Annealing*

Suhu Awal	100 - 100000
Suhu Akhir	10
Tingkat Penurunan Suhu	0.9999

- *Tabu Search* (TS)

Tabel 4.9 Parameter Pengolahan Data dengan *Tabu Search*

Jumlah Iterasi Maksimum	1000-100000
Jumlah Solusi Tetangga	100 - 200

Pengolahan data dilakukan dengan 10 replikasi, kecuali untuk problem data set yang besar yaitu OPT50, DUN62, OPT75, OPT100 dan OPT125, replikasi dilakukan sebanyak 5 kali. Replikasi ini dilakukan karena solusi yang dihasilkan berbeda-beda pada permasalahan tata letak mesin yang sama. Hal ini dimungkinkan karena solusi baru yang dihasilkan merupakan solusi random dari solusi awal yang didapatkan. Dari semua replikasi yang dihasilkan, hasil yang dipilih adalah hasil dengan nilai fungsi objektif yang paling minimum.

Tabel 4.10 Hasil Pengolahan Data dengan *Simulated Annealing*

No	Problem Set	Objective Function Value			Computation Time (s)	
		CBS - SA Best	CBS - SA Worst	CBS - SA Mean	Best Time (1 Replication)	Total Time (10 Replications)
1	KH1	168	168	168	65.390795	820.906465
2	KH2	335	335	335	52.339719	779.688186
3	KH3	360	360	360	71.755921	791.737772
4	KH4	335	335	335	79.258296	859.339255
5	KH5	14.6	14.6	14.6	77.15137	827.102024
6	KH6	267	267	267	71.910173	860.355023
7	KH7	247	247	247	57.016876	825.058633
8	KH8	42	42	42	55.608365	701.923645
9	KH9	192	192	192	52.418736	673.97353
10	NUG5	0.77	0.77	0.77	150.520405	1923.148174
11	NUG6	1.33	1.33	1.33	188.523319	2900.363282
12	NUG7	2.56	2.56	2.56	312.628232	3305.018104
13	NUG8	4.33	4.35	4.332	411.345838	6101.719034
14	NUG12	12.215	12.475	12.3725	1023.32782	11467.91306
15	NUG15	19.61	20.015	19.7869	1101.49370	11131.99457
16	NUG20	45.725	46.391	45.9553	3537.21940	54477.2107
17	OPT8	193.6	193.6	193.6	135.906406	1377.039008
18	OPT12	2626	2627.5	2626.9	2253.70652	22742.8736
19	OPT20	1146.5	1175.5	1164.7	1653.82642	27951.8166
20	OPT28	6208.29	6345.38	6307.37	3600.02808	18200.33498
21	OPT50	71925.0166	74335.1033	73040.23	18002.14235	383216.3586
22	OPT75	33931.969	34761.175	34192.28	18000.4085	152933.3444
23	OPT100	505155.661	534129.632	509984.66	14400.39969	93613.23893
24	OPT125	1018292.65	1070481.75	1050118.75	14404.66109	351837.0159
25	DUN62	3816340.3	4035867.8	3875489.21	18000.72681	692678.3437

Pengolahan data ini menghasilkan dua output, yaitu kombinasi metode *cluster boundary search* (CBS) dengan SA dan TS. Hasil kedua algoritma *meta-heuristic* ini dapat dilihat pada tabel 4.10 dan tabel 4.11. Beberapa MLP yang dioptimasi oleh SA dan TS memiliki nilai fungsi objektif yang sama, namun *layout* yang dihasilkan berbeda. *Layout* setiap MLP terbaik dapat dilihat pada lampiran kedua.

Tabel 4.11 Hasil Pengolahan Data dengan *Tabu Search*

No	Problem Set	Objective Function Value			Computation Time (s)	
		CBS - TS Best	CBS - TS Worst	CBS - TS Mean	Best Time (1 Replication)	Total Time (10 Replications)
1	KH1	168	168	168	129.12873	1295.603792
2	KH2	335	335	335	131.5369	1317.615835
3	KH3	360	360	360	131.211217	1316.736448
4	KH4	335	335	335	128.663437	1297.202028
5	KH5	14.6	14.6	14.6	129.630089	1298.458571
6	KH6	267	267	267	130.671897	1309.668437
7	KH7	247	247	247	129.187297	1294.068517
8	KH8	42	42	42	129.539893	1296.377749
9	KH9	192	192	192	132.084495	1324.222953
10	NUG5	0.77	0.77	0.77	316.604486	3186.049647
11	NUG6	1.33	1.33	1.33	531.901485	5362.64574
12	NUG7	2.56	2.56	2.56	910.710717	9195.387136
13	NUG8	4.33	4.33	4.33	1143.074644	11601.2861
14	NUG12	11.955	12.05	11.991	4001.074926	41481.69834
15	NUG15	19.235	19.57	19.3931	5000.635612	50026.29719
16	NUG20	43.58	45.64	44.9492	5000.260722	50057.06961
17	OPT8	193.6	193.6	193.6	888.901578	8999.188182
18	OPT12	2608.5	2649.5	2633.05	4000.354262	40074.91569
19	OPT20	1135	1157.5	1144.25	5000.4423	50051.0157
20	OPT28	6563.81	6941.02	6771.58	8007.507	26046.397098
21	OPT50	73166.1	74435.3711	73611.156	17573.74192	170196.6708
22	OPT75	34337.83	36648.4827	35555.037	17370.80339	18334340803
23	OPT100	514358.1	524047.079	520418.39	57614.29178	301241.7746
24	OPT125	1044123	1081494.99	1061245.49	57723.65637	303858.2197
25	DUN62	3942191	3988548.05	3969587.77	57609.85468	288349.5058

Optimasi metode CBS dan SA menghasilkan *output* nilai fungsi objektif yang cukup baik. Ada sembilan optimasi MLP yang memiliki nilai fungsi objektif terbaik sama dengan hasil penelitian terbaik sebelumnya, sebelas optimasi MLP yang memiliki nilai lebih baik daripada hasil penelitian terbaik sebelumnya dan lima optimasi MLP yang memiliki nilai fungsi objektif lebih buruk daripada hasil penelitian terbaik sebelumnya.

Hasil optimasi CBS-SA yang lebih buruk secara keseluruhan tidak lebih dari 1.7%. Untuk permasalahan DUN 62, hasil optimasi CBS-SA hampir mendekati hasil penelitian sebelumnya, dengan perbedaan tipis sebesar 0.092%.

Sedangkan untuk hasil CBS-SA dengan hasil yang lebih baik mencapai -10.92% lebih baik daripada penelitian sebelumnya, *differential evolution*, yaitu pada MLP NUG 20. Selisih perbedaan ini cukup signifikan dibandingkan dengan optimasi MLP lain yang juga lebih baik. Perbedaan yang cukup besar juga terdapat di MLP OPT 12, dimana selisih perbaikan hasil mencapai -10.037% dari hasil penelitian terbaik sebelumnya, yaitu *differential evolution*. Hasil optimasi dan perbandingan hasil terbaik dengan hasil penelitian sebelumnya dengan menggunakan CBS-SA dapat dilihat secara menyeluruh pada tabel 4.12.

Tabel 4.12 Perbandingan CBS/SA dengan Metode Lain

No	Problem Set	Machine	Best Known Solution		CBS-SA Solution	%
			Methods	Value		
1	KH1	4	Differential evolution	168.00	168.00	0.000%
2	KH2	4	Differential evolution	335.00	335.00	0.000%
3	KH3	4	Differential evolution	360.00	360.00	0.000%
4	KH4	4	Differential evolution	335.00	335.00	0.000%
5	KH5	4	Differential evolution	14.60	14.60	0.000%
6	KH6	4	Differential evolution	267.00	267.00	0.000%
7	KH7	4	Differential evolution	248.00	247.00	-0.40%
8	KH8	4	Differential evolution	42.00	42.00	0.000%
9	KH9	4	Differential evolution	192.00	192.00	0.000%
10	NUG5	5	Differential evolution	0.77	0.77	0.000%
11	NUG6	6	Differential evolution	1.36	1.33	-1.84%
12	NUG7	7	Differential evolution	2.64	2.56	-3.03%
13	NUG8	8	Differential evolution	4.37	4.33	-0.85%
14	NUG12	12	Differential evolution	13.09	12.22	-6.68%
15	NUG15	15	Differential evolution	21.09	19.61	-7.02%
16	NUG20	20	Differential evolution	51.33	45.73	-10.9%
17	OPT8	8	Differential evolution	192.50	193.60	0.571%
18	OPT12	12	Differential evolution Mckendall et al. (2009)	2918.98	2626.00	-10.0%
19	OPT20	20	(2009)	1151.40	1146.50	-0.43%
20	OPT28	28	PLANOPT Mckendall et al. (2009)	6447.25	6208.29	-3.71%
21	OPT50	50	(2009)	71291.40	71925.02	0.889%
22	OPT75	75	PLANOPT Mckendall et al. (2009)	34396.70	33931.97	1.351%
23	OPT100	100	(2009)	496820.40	505155.66	1.678%
24	OPT125	125	Mckendall et al. (2009)	1008839.0	1018292.6	
25	DUN62	62	(2009)	0	5	0.937%
			Mckendall et al. (2009)	3812825.0	3816340.3	
			(2009)	0	0	0.092%

Hasil optimasi *meta-heuristic* berikutnya adalah dengan menggunakan kombinasi CBS dan *tabu search* (TS). Output yang dihasilkan oleh CBS-TS cukup baik, dengan sembilan optimasi MLP yang memiliki nilai fungsi objektif sama dengan penelitian terbaik sebelumnya, sebelas optimasi MLP dengan nilai fungsi objektif yang lebih baik daripada penelitian sebelumnya dan lima optimasi MLP dengan nilai fungsi objektif yang lebih buruk daripada penelitian sebelumnya.

Hasil optimasi CBS-TS yang lebih buruk ada pada MLP OPT 8, OPT 50, OPT 100, OPT 125 dan DUN 62, dengan nilai rata-rata selisih perbedaan yang lebih tinggi daripada hasil optimasi dengan CBS-SA. Selisih terbesar pada optimasi CBS-TS mencapai lebih dari 3.3% dibandingkan hasil terbaik peneliti sebelumnya, Mckendall et al. (2009), yaitu pada OPT 100, OPT 125 dan DUN 62. Hasil optimasi dengan CBS-TS yang lebih baik daripada penelitian sebelumnya, *differential evolution*, mencapai selisih terbesar -15.098%, yaitu pada MLP NUG 20. MLP OPT 12 juga mencapai selisih perbedaan yang signifikan, yaitu -10.637%. Hasil optimasi dan perbandingan hasil terbaik dengan hasil penelitian sebelumnya dengan menggunakan CBS-TS dapat dilihat pada tabel 4.13.

Tabel 4.13 Perbandingan CBS/TS dengan Metode Lain

No	Problem Set	Machine	Best Known Solution		CBS-TS Solution	%
			Methods	Value		
1	KH1	4	Differential evolution	168.00	168.00	0.000%
2	KH2	4	Differential evolution	335.00	335.00	0.000%
3	KH3	4	Differential evolution	360.00	360.00	0.000%
4	KH4	4	Differential evolution	335.00	335.00	0.000%
5	KH5	4	Differential evolution	14.60	14.60	0.000%
6	KH6	4	Differential evolution	267.00	267.00	0.000%
7	KH7	4	Differential evolution	248.00	247.00	-0.403%
8	KH8	4	Differential evolution	42.00	42.00	0.000%
9	KH9	4	Differential evolution	192.00	192.00	0.000%
10	NUG5	5	Differential evolution	0.77	0.77	0.000%
11	NUG6	6	Differential evolution	1.36	1.33	-1.845%
12	NUG7	7	Differential evolution	2.64	2.56	-3.030%
13	NUG8	8	Differential evolution	4.37	4.33	-0.847%
14	NUG12	12	Differential evolution	13.09	11.96	-8.671%
15	NUG15	15	Differential evolution	21.09	19.24	-8.796%
16	NUG20	20	Differential evolution	51.33	43.58	-15.098%

Tabel 4.13 Perbandingan CBS/TS dengan Metode Lain (sambungan)

No	Problem Set	Machine	Best Known Solution		CBS-TS Solution	%
			Methods	Value		
17	OPT8	8	Differential evolution	192.50	193.60	0.571%
18	OPT12	12	Differential evolution	2918.98	2608.50	-10.63%
19	OPT20	20	Mckendall et al. (2009)	1151.40	1135.00	-1.424%
20	OPT28	28	PLANOPT	6447.25	6563.81	1.808%
21	OPT50	50	Mckendall et al. (2009)	71291.40	73166.10	2.630%
22	OPT75	75	PLANOPT	34396.70	34337.83	-0.171%
23	OPT100	100	Mckendall et al. (2009)	496820.40	514358.08	3.530%
24	OPT125	125	Mckendall et al. (2009)	1008839.00	1044123.30	3.498%
25	DUN62	62	Mckendall et al. (2009)	3812825.00	3942190.80	3.393%

4.3 Analisa Pengolahan Data

4.3.1 Analisa Hasil

1. KH 1 – KH 9

Kesembilan permasalahan tata letak mesin yang diambil dari Kusiak dan Heragu ini terdiri dari empat mesin dengan keseluruhan mesin berbentuk persegi. Sebelumnya mereka menyelesaikan MLP ini dengan menggunakan *Triangle Assignment Algorithm* (TAA), *single row*. Secara keseluruhan untuk permasalahan ini, CBS-SA dan CBS-TS lebih unggul dari pada metode TAA yang digunakan oleh Kusiak dan Heragu. Hasil yang didapatkan oleh CBS dan *differential evolution* semuanya memiliki nilai optimal yang sama.

Tabel 4.14 Perbandingan CBS-SA/TS dengan Metode Terdahulu pada Data Set KH

No	Problem	Machine	TAA Single	Best Known Solution		CBS		TAA -
			Row	Methods	Value	SA	TS	CBS
1	KH1	4	225.00	Differential evolution	168.00	168.00	168.00	-25.33%
2	KH2	4	440.00	Differential evolution	335.00	335.00	335.00	-23.86%
3	KH3	4	510.00	Differential evolution	360.00	360.00	360.00	-29.41%
4	KH4	4	465.00	Differential evolution	335.00	335.00	335.00	-27.96%
5	KH5	4	19.68	Differential evolution	14.60	14.60	14.60	-25.81%
6	KH6	4	359.00	Differential evolution	267.00	267.00	267.00	-25.63%
7	KH7	4	318.00	Differential evolution	248.00	247.00	247.00	-22.33%
8	KH8	4	60.00	Differential evolution	42.00	42.00	42.00	-30.00%
9	KH9	4	244.00	Differential evolution	192.00	192.00	192.00	-21.31%

Perbandingan antara CBS-SA dan CBS-TS dengan TAA Single Row menunjukkan perbaikan hasil yang cukup signifikan, yaitu dengan kisaran selisih sebesar 21.31% sampai 30%. Hal ini membuktikan bahwa metode CBS-SA dan CBS_TS dapat digunakan untuk menyelesaikan MLP dengan data yang kecil. Optimasi MLP KH 1 -9 memakan waktu kurang lebih satu menit, dengan tanpa adanya batasan waktu iterasi.

Kedua metode kombinasi BCS-SA dan BCS-TS tepat untuk digunakan dalam mengoptimasi permasalahan tata letak mesin berjumlah empat dengan dimensi mesin persegi. Hal ini disebabkan seluruh dimensi mesin berbentuk persegi, yang berarti bahwa jarak dari *centroid* persegi ke seluruh sisinya, yaitu atas, bawah, kiri dan kanan, sama. Keadaan ini sangat mendukung metode CBS yang mencari penempatan optimal di sekeliling seluruh mesin yang dikelilingi. Penciptaan berbagai solusi random oleh SA dan TS juga mendukung CBS untuk mencari *global optimum*.

2. NUG 5 – NUG 20

Permasalahan tata letak mesin ini diambil dari peneliti Nugent, C.E., Vollmann, T.E. and Ruml, J, 1968. Karakteristik ini sama seperti KH di mana dimensi seluruh mesin berbentuk persegi. Beberapa penyelesaian pada penelitian sebelumnya dilakukan oleh Heragu dan Kusiak dengan metode *Triangle Assignment Algorithm (TAA) single* dan *double row* untuk NUG 5 – 8. Hasil terbaik yang didapat adalah dengan menggunakan TAA double row pada NUG 5, NUG 6, NUG 7 dan NUG 8. Pada NUG 12, NUG 15 dan NUG 20 selain menggunakan TAA *single* dan *double row*, mereka juga menggunakan TAA *for cluster* MLP. Hasil penelitian mereka yang menggunakan metode TAA *for cluster* menghasilkan *output* terbaik diantara ketiga metode yang dipakai. Metode TAA *cluster* sendiri merupakan metode yang baik untuk mengoptimasi MLP dengan *multi-row*.

Tabel 4.15 Perbandingan CBS-SA/TS dengan Metode Terdahulu pada Data Set NUG

No	Problem	Machine	TAA	Best Known Solution		CBS		TAA -	DE -
				Methods	Value	SA	TS	CBS	CBS
10	NUG5	5	1.14	Differential evolution	0.77	0.77	0.77	-32.46%	0.00%
11	NUG6	6	2.01	Differential evolution	1.36	1.33	1.33	-33.83%	-1.85%
12	NUG7	7	3.98	Differential evolution	2.64	2.56	2.56	-35.68%	-3.03%
13	NUG8	8	4.95	Differential evolution	4.37	4.33	4.33	-12.53%	-0.85%
14	NUG12	12	15.77	Differential evolution	13.09	12.22	11.96	-24.19%	-8.67%
15	NUG15	15	29.09	Differential evolution	21.09	19.61	19.24	-33.88%	-8.80%
16	NUG20	20	70.86	Differential evolution	51.33	45.73	43.58	-38.50%	-15.10%

Hasil yang didapatkan dengan menggunakan CBS-SA dan CBS-TS pada data set NUG baik, dimana nilai fungsi objektif pada NUG 5 sama dengan *Differential Evolution* (DE) dan problem NUG lainnya memiliki hasil yang lebih kecil. Hasil optimasi dengan BCS-SA dan BCS-TS pada NUG 5, NUG 5, NUG 7 dan NUG 8 memiliki hasil optimum yang sama. Sedangkan untuk NUG 12, NUG 15 dan NUG 20, hasil terbaik dihasilkan oleh BCS-TS.

Perbandingan antara metode TAA oleh Kusiak dengan BCS-SA/TS memiliki selisih data yang cukup jauh, dimana perbedaan nilai fungsi objektif adalah sebesar -12.52% sampai dengan -38.5%. Sedangkan perbandingan antara DE dan BCS-SA/TS memiliki selisih sebesar 0% sampai dengan -15.1%. Hal ini menunjukkan bahwa pada data set 5 sampai 20 mesin, metode CBS-TS merupakan solusi optimasi yang paling baik. CBS menghasilkan nilai fungsi objektif yang paling optimum, dikarenakan semua mesin berbentuk persegi dan CBS melakukan pencarian yang paling optimum di sekeliling setiap mesin yang sudah diletakkan sebelumnya. Secara umum, waktu yang diperlukan untuk melakukan proses optimasi ini berkisar antara 150 sampai 5000 detik, dengan tanpa adanya batasan waktu iterasi maksimum.

3. OPT 8 – OPT 125

Problem ini menggunakan data dari *software* khusus yang digunakan untuk mengoptimasi permasalahan tata letak, PLANOPT. Data

dalam *software* ini banyak digunakan sebagai benchmark oleh peneliti di berbagai belahan dunia dalam penelitian optimasi.

Tabel 4.16 Perbandingan CBS-SA/TS dengan Metode Terdahulu pada Data Set OPT

No	Problem	Machine	PLANOPT	Best Known Solution		CBS		CBS- PLANOPT	CBS - BEST
				Methods	Value	SA	TS		
17	OPT8	8	213.50	Differential evolution	192.50	193.60	193.60	-9.32%	0.57%
18	OPT12	12	5384.43	Differential evolution	2918.98	2626.00	2608.50	-51.55%	-10.64%
19	OPT20	20	1157.00	Mckendall et al. (2009)	1151.40	1146.50	1135.00	-1.90%	-1.42%
20	OPT28	28	6447.25	PLANOPT	6447.25	6208.29	6563.81	-3.71%	-3.71%
21	OPT50	50	78224.68	Mckendall et al. (2009)	71291.40	71925.02	73166.10	-8.05%	0.89%
22	OPT75	75	34396.38	PLANOPT	34396.70	33931.97	34337.83	-1.35%	-1.35%
23	OPT100	100	538193.1	Mckendall et al. (2009)	496820.4	505155.66	514358.08	-6.14%	1.68%
24	OPT125	125	1084451	Mckendall et al. (2009)	1008839	1018292.65	1044123.3	-6.10%	0.94%

Optimasi oleh CBS-SA dan CBS-TS menghasilkan *output* yang lebih optimal jika dibandingkan dengan PLANOPT. Selisih perbandingan kedua metode ini berkisar antara -1.35% sampai dengan -10.64%. Pada optimasi dengan CBS itu sendiri, kombinasi CBS-TS menghasilkan *output* yang terbaik pada OPT 12 dan OPT 20. Sedangkan CBS-SA menghasilkan *output* terbaik pada data set yang lebih besar, yaitu OPT 28, OPT 50, OPT 75, OPT 100 dan OPT 125.

Secara keseluruhan, CBS-SA/TS menghasilkan *output* terbaik pada MLP OPT 12, OPT 20, OPT 28 dan OPT 75 dengan selisih sebesar -1.35% sampai dengan -10.64% dibandingkan dengan hasil terbaik penelitian sebelumnya, yaitu oleh *differential evolution*, PLANOPT, Mckendall et al. dan PLANOPT secara berurutan. Hasil terbaik OPT 8 dihasilkan oleh metode *differential evolution*, dengan selisih 0.57% lebih baik daripada CBS-SA/TS. Dalam hal ini, DE memang baik dalam mengoptimasi MLP dengan jumlah sedikit dan terbukti pada data set OPT 8. Selanjutnya, hasil terbaik untuk data set OPT 50, OPT 100 dan OPT 125 dilakukan oleh Mckendall et al. dengan selisih sebesar 0.89% sampai dengan 1.68% dibandingkan dengan CBS-SA.

Dari beberapa data di atas, peneliti menarik hipotesa bahwa metode yang dilakukan oleh Mckendall et al. sejauh ini memang lebih baik untuk beberapa jenis data set MLP berjumlah banyak, yaitu OPT 50, OPT 100

dan OPT 125. Namun, walaupun CBS-SA/TS kurang cocok untuk digunakan pada optimalisasi beberapa jenis data set berukuran besar, pada faktanya di penelitian ini CBS-SA menghasilkan *output* terbaik pada OPT 75. Ada faktor lain yang belum dimaksimalkan, yaitu pengoperasian program dengan waktu yang lebih lama. Walau pengoperasian program dengan waktu yang lebih lama bukan sebuah tolak ukur yang pasti untuk menghasilkan *output* terbaik, namun probabilitas ini perlu dicoba.

4. DUN 62

Permasalahan tata letak mesin ini diambil dari data penelitian Dunker, et al. yang melakukan optimasi dengan *Coevolutionary Algorithm* (CA). CBS-SA menghasilkan *output* yang lebih baik daripada CBS-TS dengan selisih -3.19% lebih baik. Hasil optimasi CBS-SA lebih buruk 0.09% daripada hasil optimasi oleh Meckendall. Namun di lain sisi, Hasil optimasi dengan CBS-SA menghasilkan *output* -9.61% lebih baik daripada hasil optimasi CA yang dilakukan Dunker.

Tabel 4.17 Perbandingan CBS-SA/TS dengan Metode Terdahulu pada Data Set DUN

Problem	Machine	Co- evolutionary Alg.	Best Known Solution		CBS		CBS - Co- evolutionary	CBS - BEST
			Methods	Value	SA	TS		
DUN62	62	4221911	Mckendall et al.	3812825	3816340.3	3942190.8	-9.61%	0.09%

Hasil penelitian pada data set DUN 62 menunjukkan bahwa CBS-SA lebih baik daripada CBS-TS untuk digunakan dalam mengoptimasi MLP dengan jumlah mesin yang banyak. Pembatasan waktu yang lebih lama untuk pengoperasian program mungkin akan memperbesar kemungkinan MATLAB untuk menghasilkan *output* yang lebih optimal

4.3.2 Analisa Waktu Pengoperasian Program

Waktu yang dibutuhkan untuk melakukan proses optimasi ini bervariasi, yaitu dari 6 detik sampai dengan beberapa hari, jika tidak ada pembatasan waktu pengoperasian MATLAB. Pada penelitian ini, pembatasan waktu maksimal pengoperasian program adalah 85.000 detik. Jika secara aktual program selesai beroperasi di atas waktu tersebut, itu merupakan faktor luar yang tidak dapat dikontrol oleh peneliti. Sejauh ini keterlambatan waktu terbesar pada pembatasan

waktu pengoperasian program adalah pada batas maksimal 85.000 detik, yaitu sebesar 1.500 detik, dalam perhitungan data set DUN 62.

Secara keseluruhan dan teoritis, waktu berpengaruh cukup signifikan dalam pengoperasian program optimasi ini, dimana waktu yang lebih lama seharusnya akan menghasilkan *output* yang lebih optimal. Dalam beberapa replikasi yang peneliti lakukan, hal tersebut memang betul, dimana ada beberapa data set berukuran besar seperti OPT 125 yang dioperasikan dengan limit waktu 78.200 detik (21.7 jam) dan menghasilkan hasil yang terbaik sebesar 1018292.65437. Namun sebaliknya, dalam beberapa keadaan ada proses optimasi yang dioperasikan lebih cepat, namun menghasilkan nilai fungsi objektif yang lebih minimal. Seperti pada data set DUN 62, dimana sejauh ini solusi optimal yang dihasilkan adalah 3816340.3 dengan limit waktu maksimum pengoperasian MATLAB 72.000 detik. Di replikasi berikutnya, optimasi pada data set yang sama dilakukan pada batas maksimal perhitungan 85.000 detik, tetapi nilai fungsi objektif yang dihasilkan lebih buruk, yaitu sebesar 3876650.

Hal ini dimungkinkan karena solusi global optimum sementara yang didapatkan oleh MATLAB terhimpit atau terjebak di antara kedua lembah. Bertepatan dengan keadaan tersebut, solusi random yang dihasilkan MATLAB tidak dapat membuat solusi global optimum sementara itu melompat keluar dari lubang tersebut untuk mencapai global optimum yang sebenarnya. Sementara MATLAB masih berusaha untuk mencari solusi random untuk mencapai global optimum, waktu iterasi yang dijalankan sudah habis.

Secara keseluruhan, peneliti menarik sebuah hipotesa bahwa, ada faktor luar yang berpengaruh dalam menghasilkan sebuah hasil yang optimal dengan waktu yang lebih cepat. Hal ini dimungkinkan ketika solusi random yang dihasilkan MATLAB pada suatu keadaan ternyata secara kebetulan menghasilkan *output* dengan nilai fungsi objektif yang terbaik. Hal ini merupakan faktor luar yang tidak dapat dikontrol oleh peneliti. Hal lain yang dapat mempercepat proses optimasi ini adalah dengan menggunakan perangkat komputer dengan spesifikasi *processor* dan *memory* yang lebih besar. Hal ini secara teoritis akan cenderung mempercepat proses pengerjaan yang dilakukan oleh komputer.

Dalam penelitian ini, waktu pengoperasian MATLAB untuk optimasi problem pada set data berukuran besar seperti OPT 50, OPT 75, OPT 100, OPT 125 dan DUN 62 dibatasi. Hal ini disebabkan lab SEMS Departemen Teknik Industri Universitas Indonesia digunakan oleh pengguna lain untuk penelitian lain di waktu yang bersamaan. Walaupun lamanya waktu bukan lah sebuah tolak ukur pasti untuk menghasilkan *output* MLP yang terbaik di samping faktor luar kinerja MATLAB yang tidak dapat dikontrol oleh peneliti, dalam penelitian ke depan waktu pengoperasian MATLAB dengan waktu lebih lama dapat dicoba untuk mendapatkan hasil yang lebih baik.

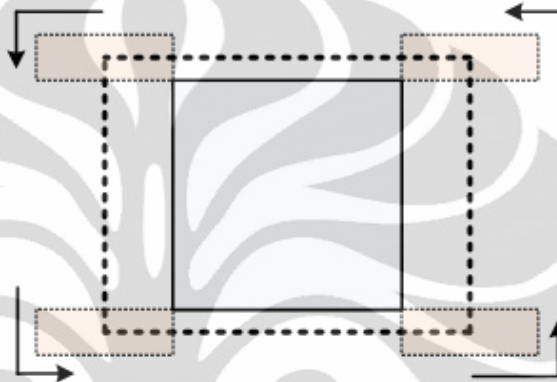
4.3.3 Analisa Metode *Cluster Boundary Search* dan Algoritma *Meta-Heuristic*

Peneliti mempelajari beberapa hal dan karakteristik optimasi di beberapa data set yang berbeda dengan metode kombinasi CBS-SA maupun CBS-TS. Hasil pengolahan data ini juga menunjukkan sejauh mana kombinasi metode ini dapat digunakan pada berbagai jenis MLP. Beberapa bagian yang dianalisa pada metode ini dapat dijabarkan sebagai berikut:

1. Optimasi tata letak mesin dengan kombinasi *cluster boundary search* dengan SA maupun TS secara umum menghasilkan *output* yang sangat baik pada data set MLP berukuran kecil, di luar OPT 8. Contoh *improvement* nilai fungsi objektif yang signifikan adalah dengan menggunakan CBS-TS pada data set NUG 20 dengan selisih perubahan - 15.1%. Sedangkan untuk MLP besar yaitu pada OPT 50, OPT 100, OPT 125 dan DUN 62, kombinasi CBS dengan algoritma *meta-heuristic* tidak menghasilkan *output* yang lebih baik. Ada pengecualian untuk data set, OPT 75, dimana metode CBS-SA dan CBS-TS menghasilkan *output* yang lebih baik daripada hasil penelitian sebelumnya, yaitu PLANOPT.
2. Metode *cluster boundary search* dan *simulated annealing* menghasilkan nilai *objective function* yang lebih baik dibandingkan dengan CBS-TS untuk menyelesaikan permasalahan tata letak mesin dengan data set berjumlah 50, 62, 75, 100 dan 125 mesin. Sedangkan metode *cluster boundary search* dan *tabu search* menghasilkan nilai *objective function*

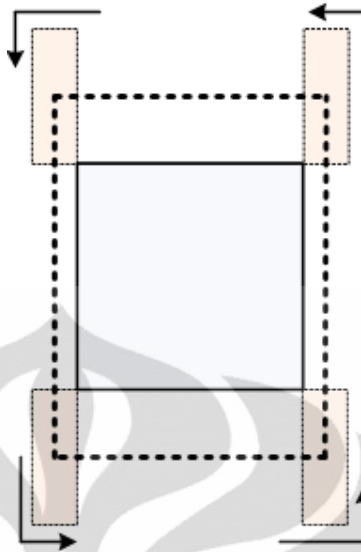
yang lebih baik daripada CBS-SA untuk menyelesaikan permasalahan tata letak mesin dengan jumlah kecil, yaitu data set dengan jumlah di bawah 50 mesin.

3. Pada MLP dengan dimensi mesin panjang dan lebar tidak sama (OPT dan DUN), ada kemungkinan untuk mendapatkan hasil optimasi yang lebih optimal. Hal ini dimungkinkan dengan melakukan proses keliling dengan orientasi pengelilingan yang berbeda.



Gambar 4.8 Proses Umum Metode *Cluster Boundary Search*

Dalam penelitian ini, mesin yang melakukan keliling tidak melakukan perubahan posisi ketika proses ini dilakukan. Panjang dan tinggi mesin yang digunakan dalam proses pengelilingan di penelitian ini sesuai dengan *input* data, sehingga mesin melakukan proses keliling dengan posisi yang sama. *Improvement* yang dapat dilakukan adalah dengan melakukan tambahan perubahan orientasi mesin yang melakukan pengelilingan. Secara umum, panjang mesin akan menjadi lebarnya dan sebaliknya. Penambahan metode ini akan membuat program ini melakukan proses keliling dioperasikan dua kali pada satu siklus keliling. Proses pertama yang dioperasikan adalah seperti pada gambar 4.8 dan proses kedua yang dioperasikan adalah seperti pada gambar 4.9.

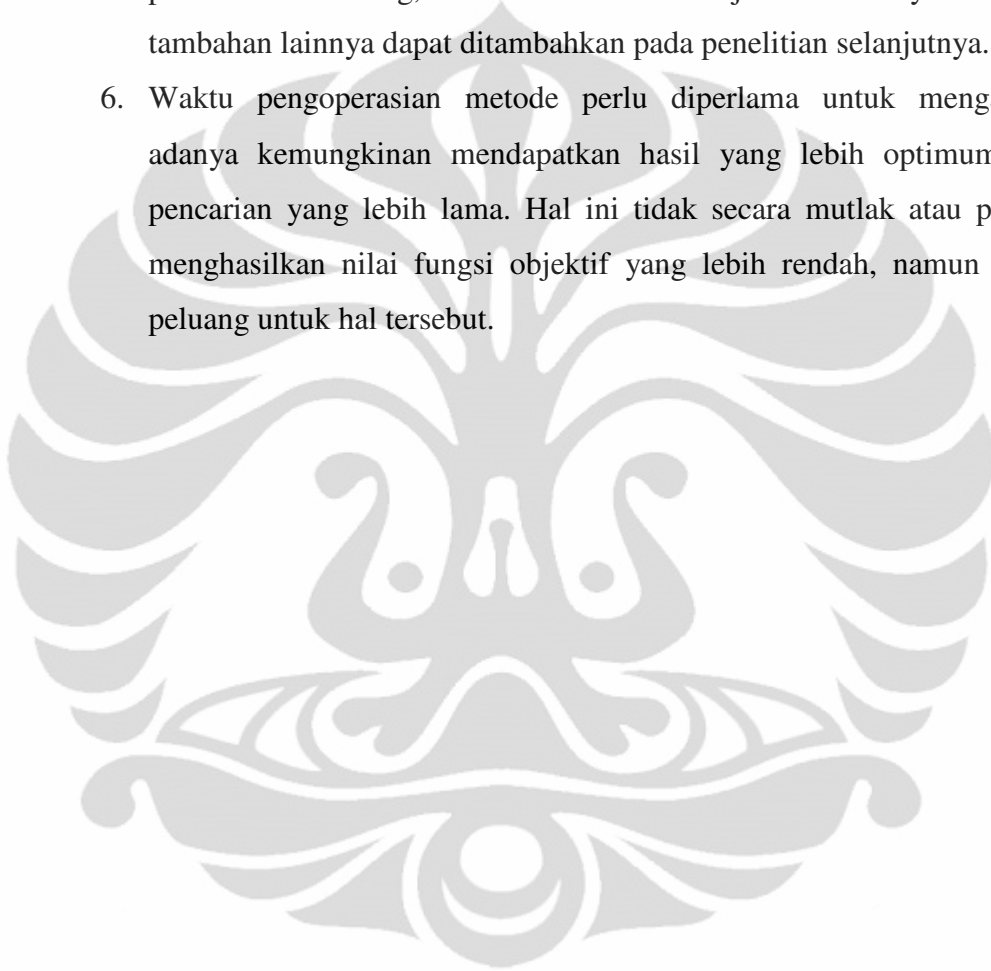


Gambar 4.9 *Improvement* pada Metode *Cluster Boundary Search*

Peneliti telah mencoba mengoperasikan metode ini dengan tambahan seperti ini dan menghasilkan *output* yang lebih baik pada data set OPT 8. Jika dengan menggunakan CBS-SA dan CBS-TS hasil optimum yang didapatkan selalu 193.6, maka dengan menambahkan perubahan arah orientasi pengelilingan hasil optimum yang didapat adalah 192.6. Ide ini mungkin dapat menjadi inspirasi atau ide bagi penelitian terkait selanjutnya.

4. Adanya keterbatasan pada MATLAB dimana MATLAB selalu memilih nilai fungsi objektif yang pertama di antara beberapa nilai optimal yang sama. Jika MATLAB dapat memilih semua nilai fungsi objektif optimal yang sama untuk kemudian diperhitungkan di proses pengelilingan selanjutnya, hal ini akan lebih dapat mengantisipasi kemungkinan solusi yang lebih optimal, secara teoritis. Kemungkinan ini akan menjadi semakin kecil ketika jumlah data set semakin besar, di mana proses pengelilingan yang dilakukan juga semakin banyak. MATLAB akan melewatkan kesempatan untuk memperhitungkan pilihan setelah pilihan pertama penempatan mesin dengan nilai optimal yang sama. Sejauh ini peneliti belum menemukan cara yang dapat dilakukan untuk mengatasi keadaan ini. Namun hal ini mungkin dapat diantisipasi pada program optimasi di penelitian selanjutnya.

5. Beberapa kendala tambahan seperti dalam kondisi *real* tata letak mesin di pabrik dapat ditambahkan. Penelitian ini hanya melakukan minimalisasi biaya *material handling*. Pada kenyataannya, ada banyak hal-hal lain yang mempengaruhi penyusunan mesin dalam sebuah sistem produksi. Hal-hal tersebut adalah *flow process* dari material, lebar dan panjang *alley* untuk pemindahan barang, faktor keamanan kerja dan lainnya. Modifikasi tambahan lainnya dapat ditambahkan pada penelitian selanjutnya.
6. Waktu pengoperasian metode perlu diperlama untuk mengantisipasi adanya kemungkinan mendapatkan hasil yang lebih optimum dengan pencarian yang lebih lama. Hal ini tidak secara mutlak atau pasti akan menghasilkan nilai fungsi objektif yang lebih rendah, namun memiliki peluang untuk hal tersebut.



BAB 5

KESIMPULAN

Setelah melakukan penelitian perancangan model optimasi tata letak mesin dengan metode *cluster boundary search* dan algoritma *meta-heuristic* dengan menggunakan *software* bahasa pemrograman MATLAB, ada beberapa kesimpulan yang dapat diambil, yaitu :

- Optimasi tata letak mesin dengan metode *cluster boundary search* dan algoritma *meta-heuristic* dengan menggunakan MATLAB R2009b telah berhasil memproses 25 permasalahan tata letak mesin dengan data yang telah dipublikasikan sebelumnya.
- Optimasi dengan menggunakan metode *cluster boundary search* dan algoritma *meta-heuristic* pada seluruh permasalahan tata letak pabrik, selain OPT 8, OPT 50, OPT 100, OPT 125 dan DUN 62, mempunyai *output* yang lebih baik, dengan range perbedaan 0% sampai dengan -15.1% dari hasil terbaik penelitian sebelumnya.
- Metode kombinasi *cluster boundary search* dan *simulated annealing* lebih cocok dipakai untuk menyelesaikan permasalahan tata letak mesin dengan jumlah besar, data set dengan jumlah 50, 62, 75, 100 dan 125 mesin.
- Metode kombinasi *cluster boundary search* dan *tabu search* lebih cocok dipakai untuk menyelesaikan permasalahan tata letak mesin dengan jumlah kecil, yaitu data set dengan jumlah di bawah 50 mesin.
- Banyak ide-ide dan improvisasi metode yang dapat ditambahkan dan dikombinasikan dengan metode *cluster boundary search*, algoritma *simulated annealing* dan *tabu search* untuk melakukan proses optimasi yang menghasilkan *output* data yang lebih baik.

REFERENSI

- Anderses, R. (2006). Solution methods to the machine *layout* problem. Thesis of Kongens Lyngby IMM Technical University of Denmark.
- Apple, J.M. (1977). *Plant Layout and Material handling*. New York: John Wiley and Sons.
- Armour, G.C., & E.S. Buffa. (1963). A heuristic algorithm and simulation approach to relative location of facilities. *Management Science*, 9, 294–309.
- Chwif, L., Barretto, R.P. and Moscato, L.A., 1998. A solution to the facility layout problem using simulated annealing. Department of Mechanical Engineering, Polytechnic Schol, University of Sao Paulo, Brazil.
- Corry, P. & E. Kozan. (2004). Ant colony optimisations for machine *layout* problems. *Computational Optimization and Applications*, 28, 287-310.
- Dunker, et al. (2003). A Coevolutionary Algorithm for A *Facility Layout Problems*. *International Journals of Production Research*, 41, 3479 – 3500.
- Engineering Optimization Software. (1996). *PLANOPT User's Manual Ver 1.50*. Florida: Deltona.
- Heragu & Kusiak. (1988). Machine *Layout* Problem in Flexible Manufacturing Systems. *Operations Research, Operations Research in Manufacturing*, 36, No. 2, 258-276.
- Imam, M.H. and Mir, M., 1998. Cluster boundary search algorithm for building-block layout optimization. *Advances in Engineering Software* Vol.29, No.2 pp. 165-173. Faculty of Engineering, Umm Al-Qura University, Saudi Arabia.
- Kusiak, A. and Heragu, S.S., 1987. The Facility Layout Problem. *European Journal of Operational Research* 29 (1987) 229-251. Department of Mechanical and Industrial Engineering, University of Manitoba, Canada
- Liggett, R.S., 2000. Automated facilities layout: past, present and future. *Automation in Construction* 197-215. Department of Architecture and Urban Design, UCLA, USA.
- Mavridou, T.D. and Pardalos, P.M., . Simulated Annealing and Genetic Algorithms for the Facility Layout Problem: A Survey. Department of Industrial and Systems Engineering, University of Florida, Gainesville, USA.

- Mckendall, A.R. and Hakobyan, A., 2009. Heuristics for the dynamic facility layout problem with unequal-area departments. Department of Industrial and Management Systems Engineering, West Virginia University, USA
- Meller, R.D. and Gau, K.Y., 1996. The Facility Layout Problem: Recent and emerging Trends and Perspectives. *Journal of Manufacturing systems* Vol.15/No.5.
- Mir & Imam. (2001). A Hybrid optimization approach for *layout* design of unequal-area facilities. *Computers & Industrial Engineering*, 39, 46-63.
- Nugent, C. E., T. E. Vollmann, & J. Ruml. (1968). An Experimental Comparison of Techniques for the Assignment of Facilities to Locations. *Operation Research*, 16, 150-173.
- Price, Kenneth V, Rainer M. Storn, & Jouni A. Lampinen. (2005). *Differential Evolution: A Practical Approach to Global Optimization*. Germany: Springer.
- Souliah, A. (1995). Simulated annealing for manufacturing systems *layout* design. *European Journal of Operational Research*, 82, 592–614.
- Tompkins, J.A., White, J.A., Bozer, Y.A. and Tanchoco, J.M.A., 2003. *Facilities Planning* (3rd ed.). John Wiley & Sons, Inc., New York.
- Yang, T. & B.A. Peters. (1998). Flexible machine *layout* design for dynamic and uncertain production environments. *European Journal of Operational Research*, 108, 49–64.
- Wong, K.Y. and Komarudin, 2009. Comparison of techniques for dealing with empty spaces in unequal area facility layout problems. Department of Manufacturing and Industrial Engineering, Faculty of Mechanical Engineering, Universiti Teknologi Malaysia
- Wong, K.Y. and Komarudin, 2010. Solving facility layout problems using Flexible Bay Structure representation and Ant System algorithm. Faculty of Mechanical Engineering, Universiti Teknologi Malaysia, Malaysia.

Set Data KH1-KH9**Ukuran Mesin**

Nomor Mesin	Panjang	Lebar
1	2	2
2	4	4
3	6	6
4	2	2

Aliran Material**KH1**

	1	2	3	4
1	0	10	5	0
2	0	0	0	20
3	0	0	0	8
4	0	0	0	0

KH6

	1	2	3	4
1	0	1	2	39
2	0	0	2	0
3	0	0	0	40
4	0	0	0	0

KH2

	1	2	3	4
1	0	10	15	15
2	0	0	0	5
3	0	0	0	40
4	0	0	0	0

KH7

	1	2	3	4
1	0	3	8	2
2	0	0	2	2
3	0	0	0	40
4	0	0	0	0

KH3

	1	2	3	4
1	0	40	40	10
2	0	0	0	0
3	0	0	0	10
4	0	0	0	0

KH8

	1	2	3	4
1	0	2	2	2
2	0	0	2	2
3	0	0	0	0
4	0	0	0	0

KH4

	1	2	3	4
1	0	10	15	20
2	0	0	10	15
3	0	0	0	10
4	0	0	0	0

KH9

	1	2	3	4
1	0	4	2	4
2	0	0	0	0
3	0	0	0	40
4	0	0	0	0

KH5

	1	2	3	4
1	0	0.3	0	0.3
2	0	0	1	0.5
3	0	0	0	1
4	0	0	0	0

Set Data NUG5-NUG20**Ukuran Mesin**

Nomor Mesin	Panjang	Lebar
1	0.04	0.04
2	0.02	0.02
3	0.05	0.05
4	0.03	0.03
5	0.01	0.01
6	0.05	0.05
7	0.08	0.08
8	0.01	0.01
9	0.05	0.05
10	0.04	0.04
11	0.05	0.05
12	0.01	0.01
13	0.03	0.03
14	0.02	0.02
15	0.03	0.03
16	0.01	0.01
17	0.02	0.02
18	0.03	0.03
19	0.04	0.04
20	0.09	0.09

Aliran Material**NUG5**

	1	2	3	4	5
1	0	0	0	0	0
2	5	0	0	0	0
3	2	3	0	0	0
4	4	0	0	0	0
5	1	2	0	5	0

NUG6

	1	2	3	4	5	6
1	0	0	0	0	0	0
2	5	0	0	0	0	0
3	2	3	0	0	0	0
4	4	0	0	0	0	0
5	1	2	0	5	0	0
6	0	2	0	2	10	0

NUG7

	1	2	3	4	5	6	7
1	0	0	0	0	0	0	0
2	5	0	0	0	0	0	0
3	2	3	0	0	0	0	0
4	4	0	0	0	0	0	0
5	1	2	0	5	0	0	0
6	0	2	0	2	10	0	0
7	0	2	5	2	0	5	0

NUG8

	1	2	3	4	5	6	7	8
1	0	0	0	0	0	0	0	0
2	5	0	0	0	0	0	0	0
3	2	3	0	0	0	0	0	0
4	4	0	0	0	0	0	0	0
5	1	2	0	5	0	0	0	0
6	0	2	0	2	10	0	0	0
7	0	2	5	2	0	5	0	0
8	6	0	5	10	0	1	10	0

NUG12

	1	2	3	4	5	6	7	8	9	10	11	12
1	0	0	0	0	0	0	0	0	0	0	0	0
2	5	0	0	0	0	0	0	0	0	0	0	0
3	2	3	0	0	0	0	0	0	0	0	0	0
4	4	0	0	0	0	0	0	0	0	0	0	0
5	1	2	0	5	0	0	0	0	0	0	0	0
6	0	2	0	2	10	0	0	0	0	0	0	0
7	0	2	5	2	0	5	0	0	0	0	0	0
8	6	0	5	10	0	1	10	0	0	0	0	0
9	2	4	5	0	0	1	5	0	0	0	0	0
10	1	5	2	0	5	5	2	0	0	0	0	0
11	1	0	2	5	1	4	3	5	10	5	0	0
12	1	0	2	5	1	0	3	0	10	0	2	0

NUG15

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
4	5	3	10	0	0	0	0	0	0	0	0	0	0	0	0
5	1	2	2	1	0	0	0	0	0	0	0	0	0	0	0
6	0	2	0	1	3	0	0	0	0	0	0	0	0	0	0
7	1	2	2	5	5	2	0	0	0	0	0	0	0	0	0
8	2	3	5	0	5	2	6	0	0	0	0	0	0	0	0
9	2	2	4	0	5	1	0	5	0	0	0	0	0	0	0
10	2	0	5	2	1	5	1	2	0	0	0	0	0	0	0
11	2	2	2	1	0	0	5	10	10	0	0	0	0	0	0
12	0	0	2	0	3	0	5	0	5	4	5	0	0	0	0
13	4	10	5	2	0	2	5	5	10	0	0	3	0	0	0
14	0	5	5	5	5	5	1	0	0	0	5	3	10	0	0
15	0	0	5	0	5	10	0	0	2	5	0	0	2	4	0

NUG20

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	5	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	10	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	5	5	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	2	1	5	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	10	5	2	5	6	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	3	1	4	2	5	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	1	2	4	1	2	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
10	5	4	5	0	5	6	0	1	2	0	0	0	0	0	0	0	0	0	0	0
11	5	2	0	10	2	0	5	10	0	5	0	0	0	0	0	0	0	0	0	0
12	5	5	0	2	0	0	10	10	3	5	5	0	0	0	0	0	0	0	0	0
13	0	0	0	2	5	10	2	2	5	0	2	2	0	0	0	0	0	0	0	0
14	0	10	5	0	1	0	2	0	5	5	5	10	2	0	0	0	0	0	0	0
15	5	10	1	2	1	2	5	10	0	1	1	5	2	5	0	0	0	0	0	0
16	4	3	0	1	1	0	1	2	5	0	10	0	1	5	3	0	0	0	0	0
17	4	0	0	5	5	1	2	5	0	0	0	1	0	1	0	0	0	0	0	0
18	0	5	5	2	2	0	1	2	0	5	2	1	0	5	5	0	5	0	0	0
19	0	10	0	5	5	1	0	2	0	5	2	2	0	5	10	2	2	1	0	0
20	1	5	0	5	1	5	10	10	2	2	5	5	5	0	10	0	0	1	6	0

Set Data OPT8 – OPT125**OPT8****Ukuran Mesin**

Nomor Mesin	Panjang	Lebar
1	2	3
2	4	5
3	2	2
4	3	3
5	2	4
6	4	4
7	4	4
8	3	4

Aliran Material

	1	2	3	4	5	6	7	8
1	0	0	0	0	0	0	0	0
2	1	0	0	0	0	0	0	0
3	2	4	0	0	0	0	0	0
4	0	3	2	0	0	0	0	0
5	0	6	0	5	0	0	0	0
6	0	0	3	2	0	0	0	0
7	2	0	1	0	0	4	0	0
8	0	2	0	2	4	0	1	0

OPT12**Ukuran Mesin**

Nomor Mesin	Panjang	Lebar
1	5	10
2	1	2
3	5	5
4	6	8
5	20	5
6	15	10
7	20	20
8	1	4
9	5	50
10	10	10
11	10	6
12	20	14

(lanjutan)

Aliran Material

	1	2	3	4	5	6	7	8	9	10	11	12
1	0	0	0	0	0	0	0	0	0	0	0	0
2	5	0	0	0	0	0	0	0	0	0	0	0
3	2	3	0	0	0	0	0	0	0	0	0	0
4	4	0	0	0	0	0	0	0	0	0	0	0
5	1	2	0	5	0	0	0	0	0	0	0	0
6	0	2	0	2	10	0	0	0	0	0	0	0
7	0	2	0	2	0	5	0	0	0	0	0	0
8	6	0	5	10	0	1	10	0	0	0	0	0
9	2	4	5	0	0	1	5	0	0	0	0	0
10	1	5	2	0	5	5	2	0	0	0	0	0
11	1	0	2	5	1	4	3	5	10	5	0	0
12	1	0	2	5	1	0	3	0	10	0	2	0

OPT20

Ukuran Mesin

Nomor Mesin	Panjang	Lebar
1	1	2
2	2	2
3	1	1
4	2	3
5	3	3
6	2	2
7	2	1
8	2	3
9	3	2
10	3	3
11	2	3
12	1	2
13	3	2
14	3	3
15	3	3
16	2	2
17	3	2
18	2	2
19	2	2
20	2	1

Aliran Material

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	0	3	0	0	4	2	0	0	4	0	0	5	3	0	5	0	0	1	0	0
2	0	0	1	0	1	2	5	0	3	0	0	0	2	0	3	0	3	1	2	3
3	0	0	0	4	0	0	3	0	0	0	1	0	0	0	0	0	5	0	2	3
4	0	0	0	0	4	0	0	1	5	3	0	2	0	0	4	5	0	1	0	0
5	0	0	0	0	0	0	0	0	1	4	1	5	0	0	3	2	0	5	0	4
6	0	0	0	0	0	0	3	0	0	5	0	0	3	0	0	0	2	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	4	0	2	0	3	2	0	1
8	0	0	0	0	0	0	0	0	0	0	2	0	0	5	0	4	0	1	0	0
9	0	0	0	0	0	0	0	0	0	3	0	5	0	0	0	2	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	5	0	1	2	4	0	3	4	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	5	5	4	0	4	3	1
12	0	0	0	0	0	0	0	0	0	0	0	0	5	0	2	0	0	1	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	2	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	5	1	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	4	3	3
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	5	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	5
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	1
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

(lanjutan)

OPT28
Ukuran Mesin

Nomor Mesin	Panjang	Lebar
1	3.5	2.5
2	2.5	4
3	2	3
4	2	3
5	2	3
6	4	4
7	4	3
8	2	3
9	3	2
10	3	3
11	2	3
12	4	2
13	3	4
14	3	3
15	3	3
16	3	4
17	4	2
18	4	4
19	4	3
20	4	3
21	3	3
22	2	2
23	3	4
24	4	3
25	3	3
26	2	3
27	3	2
28	4	4

Aliran Material

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	3	1	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	3	3	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	3	3	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	1	0	1	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	2	0	0	2	2	0	0	0	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	1	2	0	0	0	4	4	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	2	0	1	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	2	3	0	0	1	0	2	0	0	2	3	2	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	2	2	0	0	4	2	4	2	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	4	4	0	0	2	3	0	0	0	0	0	2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
18	1	1	0	1	3	0	0	1	0	3	3	1	0	2	4	3	0	0	0	0	0	0	0	0	0	0	0	0
19	0	2	2	0	0	0	0	0	0	0	2	0	0	1	2	0	1	4	0	0	0	0	0	0	0	0	0	0
20	0	3	2	0	0	0	1	0	0	0	1	0	0	0	3	0	2	1	3	0	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
22	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
23	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
24	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0
25	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	4	0	0	0	1	0	0	0	0
26	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	1	0	2	0	0	0
27	0	0	0	0	0	1	0	0	0	0	0	2	0	0	0	0	3	0	0	0	2	1	0	2	3	1	0	0
28	4	0	3	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	3	0	1	0	0	0

OPT50
Ukuran Mesin

Nomor Mesin	Panjang	Lebar	Nomor Mesin	Panjang	Lebar
1	1.031	4.182	26	2.18	1.902
2	1.045	2.759	27	2.434	4.404
3	3.568	3.765	28	1.077	1.513
4	5.081	2.281	29	4.758	3.155
5	1.97	5.584	30	5.519	3.121
6	1.294	3.976	31	2.974	5.644
7	5.228	5.574	32	3.456	1.095
8	3.724	1.451	33	3.151	4.695
9	2.414	5.057	34	4.765	4.651
10	5.287	2.305	35	4.739	3.619
11	4.179	1.236	36	5.563	4.91
12	4.592	2.68	37	5.197	2.329
13	4.89	5.8	38	3.15	5.071
14	3.386	4.938	39	5.244	1.572
15	5.375	1.669	40	1.575	3.713
16	4.615	3.461	41	4.741	5.3
17	1.31	2.856	42	2.825	1.217
18	1.119	3.575	43	4.197	1.153
19	2.217	2.451	44	3.646	4.322
20	4.949	3.602	45	1.89	4.074
21	1.097	3.74	46	2.979	3.799
22	2.305	5.037	47	1.631	3.046
23	4.68	4.44	48	5.033	5.247
24	4.106	5.848	49	1.135	5.958
25	4.307	4.063	50	2.969	3.028

OPT75
Ukuran Mesin

Nomor Mesin	Panjang	Lebar	Nomor Mesin	Panjang	Lebar
1	1.391138	3.749118	39	2.507	4.222
2	3.598932	3.804072	40	3.154	2.655
3	5.2	5.792	41	5.845	3.832
4	1.848	4.272	42	5.018	3.323
5	5.893	4.107	43	4.664	1.341
6	2.76	4.264	44	5.792	2.716
7	5.716	2.981	45	3.154	5.921
8	4.084	3.129	46	4.80708	1.562301
9	1.015	5.829	47	5.736326	5.085253
10	1.949	1.389	48	3.262	3.841
11	3.093	3.593	49	3.26	5.261
12	1.625	1.622	50	6.029139	1.555518
13	6.397042	1.746392	51	5.231	5.699
14	5.362432	5.260545	52	3.555	2.299
15	5.916	4.227	53	3.126	2.826
16	1.657	1.714	54	5.208	3.772
17	2.373	4.869	55	3.836097	6.365045
18	5.554735	3.813325	56	3.191703	3.395972
19	3.590294	4.254498	57	3.026	2.987
20	2.76509	8.754274	58	1.375	1.762
21	3.426	3.919	59	1.299	1.092
22	3.585703	5.776567	60	3.391	4.738
23	1.555	3.756	61	1.91	5.527
24	1.021	3.291	62	1.958	5.869
25	2.624	2.693	63	1.295	3.36
26	3.909	2.154	64	4.883	4.808
27	2.72229	4.153534	65	5.792	3.604
28	2.98	3.817	66	2.866	3.389
29	1.586733	5.639249	67	3.634173	3.884931
30	1.905	2.664	68	4.837	3.162
31	4.638089	3.390443	69	1.187	1.574
32	5.038293	5.869612	70	5.095	1.401
33	3.458	5.265	71	2.85547	7.055867
34	4.387204	5.124254	72	4.68	4.592
35	3.031673	3.674388	73	3.126	4.064
36	2.779	1.848	74	2.975976	4.234814
37	5.793	3.939	75	1.351	3.631
38	1.098	4.752			

(lanjutan)

Aliran Material (bagian 1)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38		
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	1	0	0	2	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	2	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	1	0	0	3	0	4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	5	0	0	0	2	3	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	1	1	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	1	0	4	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	1	0	4	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	3	0	0	0	2	0	0	0	2	1	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	4	1	1	1	0	1	0	1	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	1	0	5	0	0	1	0	5	0	4	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	2	1	0	5	0	0	0	0	0	0	0	0	0	4	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	5	0	0	0	3	0	0	0	0	1	0	4	0	0	0	5	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	2	3	1	0	0	2	1	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
21	0	3	0	0	0	0	3	0	0	0	0	4	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	1	0	0	1	0	0	0	0	4	2	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	5	0	0	3	3	0	4	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	1	0	0	1	5	0	0	4	0	0	0	0	5	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	3	0	0	0	5	0	0	0	0	3	0	0	0	0	3	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
26	0	3	5	0	0	4	2	5	3	1	1	0	0	4	1	0	0	1	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	3	0	2	0	0	0	0	0	0	0	0	3	0	0	0	0	5	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
28	1	1	0	0	0	1	4	0	1	0	5	1	0	0	0	0	0	3	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
29	5	2	0	5	5	0	0	1	3	0	0	0	3	0	0	0	3	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	1	2	0	0	4	0	0	0	0	0	1	2	0	0	0	0	0	4	5	3	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
31	0	4	0	0	3	0	1	0	0	0	0	0	0	0	0	0	0	3	0	0	1	0	1	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0
32	0	0	0	0	0	0	0	0	0	3	0	4	0	2	0	1	4	4	0	0	2	3	0	2	0	5	0	2	0	0	0	0	0	0	0	0	0	0	0	0
33	0	2	4	0	0	3	0	0	4	1	0	0	3	4	0	1	1	0	1	4	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
34	2	0	0	4	0	0	5	0	1	0	0	3	0	0	0	0	5	0	0	0	0	0	0	4	1	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0
35	0	0	0	1	5	0	3	0	0	0	1	2	5	0	0	1	0	4	0	1	0	0	0	4	0	5	0	1	0	0	3	0	0	0	0	0	0	0	0	0
36	4	0	0	0	5	0	0	0	0	0	5	1	0	0	4	0	0	0	0	1	4	0	0	1	0	0	2	5	0	0	1	2	0	0	0	0	0	0	0	
37	0	0	0	0	0	0	0	0	0	0	1	0	1	0	5	0	0	0	0	0	4	4	3	0	1	0	0	0	0	1	3	0	3	3	0	0	0	0	0	
38	1	0	0	0	0	3	0	4	0	2	1	0	4	0	0	2	0	5	0	0	2	0	1	1	0	1	0	1	0	0	0	0	1	0	0	5	0	0	0	

(lanjutan)

Aliran Material (bagian 3)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38
39	0	0	5	4	0	2	0	0	1	1	0	0	0	2	5	0	0	0	2	0	0	3	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0
40	0	0	0	2	0	0	0	0	3	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	3	0	0	0	0	0	0	0	0
41	0	3	0	0	0	2	0	3	0	0	2	3	1	0	0	0	0	3	4	0	0	3	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0
42	1	2	0	3	0	2	0	0	0	0	0	1	0	5	0	0	0	0	3	3	0	0	3	0	0	0	0	2	0	1	0	0	0	2	4	0	0	0
43	0	0	0	0	1	0	4	0	0	1	0	0	1	0	0	0	0	0	0	5	0	0	0	0	0	0	2	1	2	0	0	5	2	5	0	0	0	5
44	5	1	1	0	0	0	0	3	3	4	0	0	0	0	0	0	0	0	0	0	5	4	0	1	0	5	1	0	0	0	5	0	0	3	0	0	1	
45	0	0	0	0	0	1	0	0	0	0	0	0	0	3	0	1	0	5	0	0	5	0	4	0	0	0	0	0	2	0	0	0	0	0	2	0	0	
46	0	0	0	3	2	5	0	0	1	0	5	0	0	3	2	1	0	0	4	0	1	0	0	0	4	0	0	2	5	2	0	0	0	5	0	0	2	
47	0	1	0	0	0	5	0	0	1	0	0	5	0	3	0	0	1	3	0	4	0	1	0	2	0	0	1	0	0	0	0	5	0	0	0	0	1	
48	0	0	0	0	0	0	0	0	0	4	1	5	1	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	3	1	2	0	0	0	0	0	0	
49	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	5	0	0	0	0	3	0	0	0	0	0	4	0	0	0	5	0	0	0	
50	0	0	3	5	0	1	0	0	0	0	0	1	0	0	2	0	0	0	5	4	0	0	0	0	0	0	4	0	0	0	1	5	1	0	0	1		
51	2	3	0	2	0	0	1	1	1	0	0	0	1	2	4	0	1	0	5	0	0	0	0	0	5	0	4	0	0	0	0	0	2	0	0	0	0	
52	0	0	2	2	0	1	0	3	0	0	0	5	0	0	0	0	5	0	0	0	0	0	1	0	0	0	0	5	0	1	0	4	1	4	0	0	0	
53	0	0	1	0	5	0	0	0	0	2	0	0	0	0	0	4	1	0	5	2	0	0	0	1	0	0	0	3	0	1	2	0	0	0	0	0	0	
54	0	0	0	0	0	0	5	3	0	0	0	1	0	0	2	3	3	0	0	2	0	0	0	0	0	1	0	0	0	1	0	2	0	0	0	0	0	
55	2	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	2	0	0	2	0	0	1	2	0	0	0	0	0	0	3	0	4	0	2	0	
56	0	0	0	0	0	2	0	3	1	0	0	0	0	0	0	0	0	0	0	4	1	0	0	3	0	0	0	0	3	0	0	0	0	0	0	0	0	0
57	0	0	0	0	0	0	2	0	5	0	4	0	0	0	0	0	4	0	5	0	2	0	0	1	3	0	0	0	1	0	4	0	0	0	5	0	0	0
58	1	4	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	4	3	0	0	3	0	0	4	0	0	0	0	0	0	0	0	0	0
59	0	4	0	1	0	0	0	1	0	0	0	4	1	0	0	0	0	0	0	0	1	1	3	5	0	0	0	0	0	1	0	0	0	5	0	0	0	0
60	1	0	0	4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0	1	0	5	0	0	0	0	1	
61	0	4	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	3	0	0	0	1	0	0	
62	0	0	0	2	4	0	0	0	3	2	4	0	0	0	0	0	3	0	5	1	3	0	0	0	0	0	0	0	4	0	4	0	0	0	0	0	0	0
63	0	0	0	1	0	0	0	0	0	0	5	0	0	3	5	1	0	0	0	0	1	0	0	2	5	0	5	0	0	2	0	0	0	0	0	0	1	0
64	4	0	3	0	2	4	0	0	0	5	0	4	5	0	3	0	0	0	1	0	0	0	3	5	0	0	0	0	1	0	1	0	0	1	0	0	1	0
65	2	0	0	1	0	0	1	0	0	0	0	2	2	0	3	1	0	4	1	0	0	3	0	0	5	0	1	4	0	0	0	0	0	0	0	0	0	0
66	0	5	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	5	4	2	0	1	1	0	0	0	3	5	0	3	0	1	0	0	
67	1	0	0	0	0	0	0	4	5	0	0	2	0	0	0	1	0	4	0	1	0	0	0	1	0	1	0	0	5	0	0	0	0	0	0	0	0	0
68	0	0	0	0	5	0	0	2	3	0	3	0	0	0	1	0	1	0	5	0	0	0	4	1	0	1	0	0	5	1	0	0	0	0	0	0	0	0
69	1	2	0	1	3	1	0	1	0	0	0	3	0	0	2	0	0	0	3	1	0	0	4	0	0	1	0	1	0	0	0	1	3	0	0	0	0	0
70	5	5	1	0	0	0	3	0	0	0	0	1	0	0	0	2	0	0	0	2	0	4	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0
71	0	5	0	0	0	0	1	0	4	0	0	0	1	0	0	5	1	0	0	0	2	0	0	4	0	0	0	5	2	2	0	5	0	0	2	0	4	0
72	0	5	0	0	0	0	0	3	0	2	0	0	0	0	3	3	0	0	0	2	0	0	0	0	0	0	0	1	0	3	0	1	0	0	0	0	5	0
73	0	0	0	0	0	0	3	3	5	0	0	2	0	0	5	0	2	2	0	0	4	0	0	1	0	0	5	4	2	0	0	0	1	1	1	0	0	0
74	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	2	0	0	0	0	0	0	0	0	0	3	0	1	0	0	0	1	0	3	5	0	0
75	2	0	5	0	0	0	1	0	0	1	0	0	0	0	0	0	2	0	3	4	0	0	0	0	0	4	0	0	0	0	5	2	0	0	0	1	4	0

(lanjutan)

Aliran Material (bagian 4)

	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	
39	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
40	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
41	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
42	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
43	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
44	4	0	1	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
45	0	0	5	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
46	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
47	0	0	2	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
48	0	1	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
49	1	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
50	0	0	0	4	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
51	5	0	0	4	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
52	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
53	0	0	1	0	0	0	0	0	5	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
54	0	0	5	0	5	2	0	2	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
55	0	0	1	0	0	3	0	0	0	0	0	4	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
56	0	0	0	0	0	0	4	0	3	0	0	0	2	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
57	1	0	0	0	5	0	0	5	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
58	0	0	0	3	1	0	3	0	3	1	0	0	1	0	0	3	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
59	0	0	0	1	0	0	1	5	0	0	0	4	0	0	0	4	0	1	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
60	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
61	0	0	5	2	5	0	0	0	4	0	5	0	0	3	0	0	0	0	2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
62	0	0	0	0	0	1	0	5	0	0	0	3	0	0	4	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
63	0	0	0	5	0	0	5	0	0	1	0	0	0	0	0	0	4	5	0	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
64	0	0	4	0	2	0	1	4	0	0	1	0	0	1	0	0	0	0	0	5	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
65	0	0	0	0	0	5	0	0	0	5	0	0	0	0	0	0	5	5	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
66	0	0	0	0	0	0	0	0	2	0	0	0	5	0	0	1	0	0	3	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0
67	0	0	2	0	0	0	0	4	0	0	0	0	0	0	5	1	0	3	0	0	0	0	0	1	2	1	0	0	0	0	0	0	0	0	0	0	0	0
68	1	0	0	0	0	2	3	0	1	0	0	1	0	0	0	0	2	0	4	0	0	0	0	4	1	1	0	0	0	0	0	0	0	0	0	0	0	0
69	0	0	0	3	1	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	3	2	0	0	0	0	0	0	0	0	0
70	0	0	5	0	0	0	0	0	5	5	0	0	0	3	1	0	2	0	4	0	0	0	0	0	0	3	0	2	0	0	0	0	0	0	0	0	0	0
71	0	0	0	0	0	0	4	0	0	2	0	0	0	4	0	0	0	0	0	0	0	3	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0
72	5	5	0	0	0	4	1	2	0	4	0	0	0	3	0	0	4	0	2	5	0	0	0	0	0	0	0	0	0	1	0	4	3	0	0	0	0	
73	0	0	0	1	4	0	1	0	3	0	0	2	0	5	1	2	0	2	0	0	0	1	0	4	0	5	0	0	0	4	0	2	0	0	0	0	0	0
74	5	0	4	0	1	5	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	4	1	0	5	0	0	0	0	0	5	0	0
75	1	0	3	0	0	0	0	1	1	0	0	0	0	0	0	0	0	4	0	1	0	3	1	0	0	0	0	0	1	0	4	0	0	0	1	0	0	

OPT100
Ukuran Mesin

Nomor Mesin	Panjang	Lebar	Nomor Mesin	Panjang	Lebar
1	1.0557	1.1855	51	5.6634	5.0723
2	4.739	2.7172	52	5.8541	2.727
3	4.5414	1.8589	53	4.9803	2.3794
4	3.9868	3.5197	54	5.4922	4.1122
5	1.575	1.1573	55	2.9962	3.267
6	3.3874	2.0866	56	3.2548	2.5694
7	1.6401	4.9455	57	1.696	1.7234
8	4.9644	1.9767	58	1.2147	2.1346
9	3.699	1.2785	59	1.5458	1.9151
10	5.971	1.6839	60	5.9911	3.829
11	2.8518	5.422	61	3.7798	1.3323
12	3.3634	1.219	62	3.8474	5.3663
13	2.6225	2.9957	63	3.3091	2.058
14	3.8981	1.8159	64	1.4138	3.9851
15	2.8684	1.2438	65	2.257	5.1624
16	4.8547	3.765	66	2.0349	2.405
17	3.3158	3.4294	67	4.1203	3.2232
18	5.3834	2.3692	68	2.7426	2.9096
19	2.7198	1.3531	69	3.5528	5.9692
20	3.4782	2.0262	70	1.8534	4.5532
21	3.6495	5.9988	71	1.7768	4.0501
22	3.5737	3.4355	72	2.1914	3.4922
23	2.8985	1.3026	73	2.6177	5.9315
24	4.7683	3.9308	74	4.9357	1.4436
25	2.5198	1.0473	75	4.1641	2.9139
26	4.6719	5.9364	76	4.0933	2.2651
27	1.4854	3.8821	77	1.0529	2.2775
28	3.8781	3.55	78	1.5963	1.6639
29	2.9728	1.4668	79	2.3802	3.5514
30	1.0627	2.2491	80	3.0253	5.0279
31	2.2738	4.7421	81	5.2351	2.9087
32	3.7054	1.7228	82	3.9562	3.3822
33	5.3306	3.3232	83	3.8612	5.6857
34	4.438	1.8301	84	1.9338	2.4728
35	2.8951	3.0198	85	4.441	2.026
36	3.0166	1.6805	86	4.0685	4.9714
37	5.3152	1.481	87	5.5061	5.651
38	3.9204	1.8418	88	5.2595	5.8097
39	2.9437	1.9639	89	3.9729	5.7536
40	4.5727	4.6719	90	2.0423	5.6346
41	2.1743	5.2288	91	1.8017	2.0181
42	4.1898	2.7761	92	2.0617	2.3911
43	2.1015	4.5588	93	3.2833	2.7648

(lanjutan)

44	3.2534	5.5886	94	4.1688	1.0296
45	1.806	3.9039	95	2.9479	1.2191
46	3.6077	4.0803	96	4.1442	1.3372
47	3.2638	4.1508	97	4.1006	4.9067
48	4.1828	3.5897	98	1.0456	3.8949
49	4.8832	2.5175	99	2.9821	3.1724
50	2.3419	4.7814	100	4.1801	4.1602



Aliran Material (bagian 3)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	
51	5	8	7	6	6	7	6	6	5	9	8	10	6	4	6	7	7	4	1	5	6	5	5	6	6	9	2	3	8	5	9	7	4	4	5	7	5	6	6	5	6	7	3	3	10	5	8	6	3	3	
52	8	6	6	5	1	4	5	4	7	2	6	8	4	9	10	7	8	4	10	3	5	4	5	8	5	10	7	9	7	6	7	6	6	2	7	8	4	2	5	5	5	7	3	8	6	9	8	6	2	5	
53	3	3	10	3	8	3	7	5	4	5	2	5	7	4	6	3	7	5	6	6	6	9	4	5	8	4	7	8	2	6	7	2	8	6	7	3	7	7	4	2	3	6	4	7	9	6	5	4	3	4	
54	7	4	7	5	7	3	10	3	6	6	5	4	7	7	7	9	3	6	7	7	7	6	9	2	8	8	7	4	9	7	9	3	8	7	4	5	3	5	3	4	9	5	7	6	5	6	7	2	6	4	
55	7	4	6	4	8	6	6	5	3	6	4	5	9	7	9	8	7	3	9	5	8	2	7	7	4	3	6	2	9	5	7	7	9	5	5	4	7	4	7	6	8	5	4	4	2	7	4	7	5	8	
56	7	4	4	8	7	5	4	3	3	4	3	8	3	5	10	4	6	5	3	2	8	4	6	6	9	7	4	4	6	1	6	2	5	6	8	2	9	4	8	5	7	3	6	3	3	5	6	7	2	8	
57	4	5	3	6	5	8	8	6	4	10	2	4	6	3	3	4	4	7	5	7	5	4	2	5	5	3	6	6	5	5	7	6	8	3	8	6	5	5	3	7	2	5	4	2	7	6	3	9	4	7	
58	7	6	5	2	7	5	8	3	1	6	8	7	5	4	5	4	1	9	6	4	1	5	4	1	4	6	2	6	6	4	3	4	2	6	4	9	4	5	4	6	7	8	7	7	3	4	5	5	7	5	
59	3	8	8	7	4	7	4	5	6	5	8	5	10	7	5	5	7	8	9	3	4	3	7	3	4	6	4	8	2	5	2	4	7	2	8	7	6	5	6	4	4	3	7	9	2	4	7	5	6		
60	9	7	7	5	7	6	4	4	3	4	5	4	6	7	9	6	5	5	7	5	5	7	5	2	4	6	10	10	5	5	5	5	8	5	7	5	7	4	7	5	5	6	6	3	4	4	6	2	8	4	
61	9	5	4	7	3	9	2	4	4	6	6	5	4	4	5	3	10	6	4	5	4	7	9	2	5	7	3	5	5	4	8	8	4	3	7	8	7	6	2	6	9	3	6	6	8	3	4	5	3	3	
62	6	7	7	8	4	6	6	7	4	7	3	7	5	7	6	2	5	6	9	5	8	5	9	4	6	10	5	5	9	6	10	5	6	5	7	5	3	7	4	4	4	3	5	5	4	8	3	8	5	7	
63	7	3	3	5	7	7	5	8	5	4	6	8	5	4	6	1	3	8	2	6	4	8	9	5	6	7	8	3	4	6	6	6	8	4	8	4	9	6	6	3	4	5	6	6	6	4	7	6	5	7	
64	4	5	7	5	3	9	7	4	5	4	6	8	5	3	6	4	5	5	6	5	6	4	5	4	7	2	4	6	6	4	6	5	5	6	3	1	4	3	6	4	6	9	5	4	3	9	5	5	5	4	
65	2	2	7	4	5	7	7	8	4	5	4	5	4	5	4	3	8	5	7	5	3	4	3	5	5	7	4	7	2	7	6	7	5	5	4	10	5	7	8	6	7	4	8	4	5	6	6	4	7	4	
66	7	7	5	6	4	7	2	4	6	2	5	1	5	2	5	5	5	6	8	3	6	7	8	2	8	6	7	5	4	7	9	5	4	5	6	6	4	9	7	3	6	8	7	5	4	6	5	8	6	2	
67	6	6	9	5	2	4	2	5	3	3	9	8	5	5	8	7	6	6	2	5	7	6	4	5	9	4	8	9	4	3	6	4	4	4	6	6	1	6	7	5	4	7	6	9	2	4	5	8	6	4	
68	4	6	5	8	6	10	8	6	6	3	3	10	6	8	7	9	5	10	5	6	10	1	1	6	8	3	7	8	7	9	7	7	7	7	5	6	8	3	7	5	7	8	5	8	8	2	4	6	8	10	
69	2	7	7	3	5	6	4	2	6	9	4	7	3	5	5	6	3	2	2	5	7	7	3	4	6	7	4	6	6	2	6	4	4	8	5	1	4	3	5	5	2	6	5	4	6	5	8	6	9	6	
70	1	4	6	2	6	3	2	9	3	6	6	5	9	5	10	6	4	6	6	8	7	7	6	4	7	4	8	4	5	5	3	4	5	6	4	6	3	3	5	5	5	6	5	4	5	7	4	2	4	6	
71	4	6	5	8	9	1	5	8	9	5	4	4	8	5	6	2	6	6	6	5	8	8	5	9	3	6	3	4	7	4	9	8	3	4	7	8	5	8	7	2	7	8	8	4	8	5	1	7	3	9	
72	6	6	5	10	6	7	4	9	3	2	4	5	2	3	8	3	10	5	5	5	8	8	8	3	5	4	3	9	8	1	5	3	7	7	5	7	4	9	3	6	3	9	4	4	2	6	4	6	5		
73	2	2	7	3	5	5	4	6	5	4	2	4	2	7	7	8	5	5	7	3	7	6	3	3	6	6	6	5	10	7	4	3	2	3	6	5	4	1	7	5	5	7	9	8	8	7	5	3	5		
74	7	8	6	5	5	3	9	8	4	5	8	6	4	5	4	8	10	2	10	5	8	6	7	4	3	4	6	8	8	4	10	4	8	2	7	2	5	8	8	5	6	5	8	4	10	5	8	6	7	3	
75	3	9	5	8	5	3	7	10	3	4	3	8	5	5	2	9	7	3	4	4	5	5	7	8	3	3	6	9	4	6	5	2	1	9	3	7	6	6	3	7	4	5	3	8	4	7	9	7			
76	6	2	5	6	8	6	4	6	5	7	5	6	2	6	2	4	6	6	5	5	4	5	7	6	5	8	4	6	7	6	9	5	3	3	3	7	7	5	9	7	5	7	7	5	8	5	4	7	7		
77	5	4	5	4	4	5	9	5	6	3	3	6	6	2	2	3	8	2	8	6	5	3	6	7	7	4	7	5	3	1	3	5	7	8	3	5	8	9	7	3	8	2	4	8	7	7	4	5	7	5	
78	9	5	5	7	3	5	5	5	8	4	5	6	5	4	6	8	2	4	5	7	5	4	3	8	5	7	6	6	4	7	4	5	6	6	7	5	4	5	6	7	5	2	3	4	3	1	9	7	2	6	
79	5	5	4	7	7	8	9	6	4	4	2	5	7	7	8	6	4	4	8	9	3	4	5	3	1	9	8	5	6	3	1	6	9	5	6	6	5	6	5	3	5	2	8	4	4	6	5	4	10	4	
80	7	4	7	3	5	5	3	3	4	7	3	6	9	4	9	5	3	3	6	5	8	2	5	3	4	6	6	6	8	7	6	8	5	4	6	6	4	8	4	3	8	8	9	2	3	3	4	4	9	4	
81	6	5	8	3	6	9	8	3	8	6	8	6	3	3	3	7	8	5	4	5	6	7	5	2	9	6	7	4	8	5	7	6	6	9	5	4	5	4	8	5	2	5	3	5	2	7	7	3	10	1	
82	7	7	7	4	7	7	1	5	7	4	5	5	5	3	7	6	6	4	3	6	5	7	6	4	5	4	8	5	2	2	6	4	5	5	8	8	7	8	9	9	5	6	7	4	6	8	5	5	3		
83	7	3	7	6	6	6	2	7	6	7	4	1	3	2	3	4	4	5	6	5	3	5	6	2	8	7	7	4	4	6	9	3	6	5	6	8	5	5	5	3	5	8	9	7	7	5	7	3	8	3	
84	6	1	8	5	2	1	5	5	6	3	10	9	8	1	5	4	3	5	5	10	4	5	6	7	6	8	7	7	4	4	7	5	6	4	7	7	5	2	10	5	2	5	5	5	8	4	5	8	1	8	
85	3	7	9	3	7	7	9	7	5	6	7	9	4	5	3	7	6	2	6	7	3	2	3	9	6	6	10	4	8	4	6	6	4	8	6	2	3	8	7	6	4	5	2	5	8	7	7	5	3		
86	9	5	2	5	6	4	4	6	6	4	2	6	3	6	8	2	5	4	4	6	8	8	8	7	5	5	8	3	3	4	2	3	5	5	2	9	6	7	6	6	7	2	3	6	7	7	4	5	10	9	
87	6	3	3	8	2	8	5	4	2	8	2	9	7	4	7	7	8	4	1	6	5	9	4	3	7	5	8	3	4	4	6	4	3	5	4	3	6	2	4	3	6	3	7	5	4	8	6	5	9	6	
88	7	4	2	6	2	5	8	5	5	7	6	6	3	8	2	6	5	8	2	5	8	6	4	5	3	6	6	5	6	8	5	5	3	2	7	6	7	8	3	8	2	10	5	6	7	4	6	3	2	3	6
89	2	8	8	2	6	6	1	7	8	4	8	4	8	4	4	4	7																																		

OPT125
Ukuran Mesin

Nomor Mesin	Panjang	Lebar	Nomor Mesin	Panjang	Lebar
1	1.0063	3.8178	64	4.7036	4.0602
2	3.925	3.3993	65	5.0225	1.7455
3	5.1141	4.7329	66	5.5576	4.0734
4	4.5524	3.5676	67	4.3388	5.8825
5	1.457	2.8222	68	2.5291	1.8696
6	5.9425	3.2284	69	5.256	4.7215
7	1.0446	2.8893	70	1.3967	1.383
8	4.0087	4.0357	71	3.7254	3.2412
9	3.2539	2.7606	72	3.3277	3.506
10	4.9165	5.0129	73	4.6899	2.5694
11	5.3797	4.6333	74	5.3666	4.625
12	3.6967	1.7117	75	1.6361	1.3287
13	5.3111	2.048	76	4.0481	5.7804
14	5.9838	5.9983	77	4.2692	2.6106
15	2.331	2.4864	78	2.1354	2.4514
16	2.8793	1.4631	79	4.3139	1.5727
17	1.0439	5.5938	80	3.484	4.9667
18	3.9395	4.4558	81	4.4407	3.6607
19	3.4246	2.0268	82	1.0294	4.5393
20	3.2897	5.7456	83	5.3161	3.4574
21	3.9951	2.9261	84	2.9005	4.9267
22	3.8619	2.8066	85	5.7784	4.1541
23	3.1257	5.0143	86	1.6581	4.7163
24	4.7576	2.7278	87	1.1392	2.6492
25	3.4594	1.3177	88	1.6581	5.2352
26	1.7375	5.7478	89	4.6081	5.2697
27	4.4644	2.5152	90	4.5394	4.0856
28	5.8329	4.4158	91	1.8446	4.1204
29	5.1083	3.9102	92	2.8378	4.3049
30	5.0858	3.3763	93	3.6326	4.0554
31	4.66	3.0279	94	1.724	4.1508
32	4.4111	4.7791	95	1.6827	5.2758
33	1.6151	2.839	96	3.8667	2.5114
34	3.585	4.3148	97	2.5567	1.5531
35	5.7466	5.6068	98	2.4212	4.9404
36	3.3586	2.8748	99	4.7189	4.076
37	3.2804	2.3594	100	2.1424	5.3178
38	4.6958	3.8363	101	3.7119	5.924
39	5.1971	2.9882	102	3.6233	3.1339
40	1.1373	5.973	103	5.4576	2.1638

(lanjutan)

41	3.6566	1.9703	104	5.6581	1.4005
42	4.288	1.9892	105	2.682	5.5734
43	1.5496	4.7155	106	5.7307	5.1858
44	2.4304	2.6815	107	4.4676	2.9884
45	5.173	4.5399	108	3.6279	5.7739
46	2.2636	1.7224	109	3.9277	2.2756
47	5.0311	5.263	110	3.1774	5.451
48	3.766	1.0713	111	4.0077	4.9307
49	4.761	4.4306	112	2.1116	2.915
50	3.1835	2.0097	113	1.4112	4.2996
51	3.1834	2.1621	114	5.0552	4.3102
52	4.1433	1.8009	115	3.6506	4.428
53	4.4787	5.6239	116	4.6394	4.8885
54	1.8917	5.9757	117	4.2225	4.5327
55	1.4875	4.1258	118	5.7394	1.2939
56	5.6574	1.2422	119	5.9088	4.0998
57	2.1365	4.8452	120	2.8376	4.4726
58	4.1403	4.0206	121	2.2027	3.6071
59	3.989	4.1735	122	5.5131	3.2086
60	4.1238	4.6044	123	1.8589	5.8718
61	1.9213	4.6894	124	2.0532	3.283
62	2.2143	1.9447	125	1.5702	3.0235
63	3.923	2.7564			

Aliran Material (bagian 3)

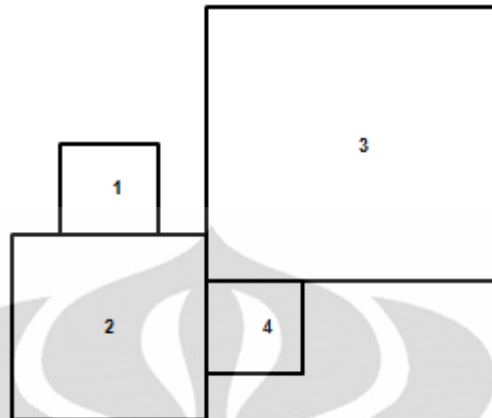
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63			
64	3	6	5	2	5	4	2	5	6	4	5	8	5	4	6	8	9	8	9	8	9	8	9	8	5	9	5	9	5	6	3	6	7	2	4	4	6	7	8	3	7	4	5	4	7	7	4	5	6	4	6	7	4	8	7	7	7	2	5	3	4	6	3	8		
65	3	4	4	3	6	4	2	6	6	6	5	6	9	4	6	2	9	4	8	7	10	3	4	6	5	7	7	6	7	2	5	9	6	7	7	5	6	8	1	6	5	7	7	5	6	6	9	7	5	4	4	2	7	2	2	4	4	8	6	4	6	8	7			
66	5	5	9	8	1	3	4	6	2	6	9	3	7	7	9	7	4	7	8	5	5	6	4	2	6	7	7	6	3	7	6	9	4	2	9	7	9	10	6	2	9	6	2	6	9	6	4	6	3	5	8	5	4	8	7	7	6	2	5	8	4	8	5	4	6	9
67	7	6	7	2	7	5	8	6	3	2	4	5	6	9	6	5	7	8	7	9	4	5	3	6	7	5	4	3	8	1	10	3	4	4	7	7	6	5	9	3	8	9	6	5	6	9	2	8	7	7	2	5	6	8	4	7	5	7	8	5	4					
68	4	7	6	2	5	7	6	4	4	6	5	6	4	5	1	3	7	10	2	5	8	4	7	1	4	3	6	7	7	3	4	9	4	6	8	6	8	5	3	7	4	5	7	4	5	8	6	4	2	5	9	6	1	8	4	8	6	3	9	4						
69	9	2	7	4	3	4	8	2	7	8	7	8	5	7	4	6	7	1	9	6	8	6	5	7	6	6	4	2	5	6	9	4	4	7	10	4	6	9	5	6	9	7	6	9	8	2	4	6	7	3	4	6	8	4	9	9	8	8	4	6	7	3	5			
70	9	3	6	4	7	7	2	3	8	5	3	6	6	3	6	6	7	8	5	6	4	5	5	8	6	7	7	6	10	5	6	6	9	8	5	4	1	5	5	3	6	6	3	2	6	9	7	2	7	3	7	2	7	4	6	6	4	4	2	2	3	10	7			
71	4	8	7	7	9	4	6	7	4	8	3	6	6	5	5	6	4	6	7	8	3	6	8	10	2	2	4	7	7	5	4	6	6	6	6	2	2	4	3	1	7	6	2	3	5	4	4	8	4	10	5	6	5	10	4	5	1	4	4	7	4	4				
72	7	8	2	2	4	3	5	5	3	6	7	3	5	9	7	5	6	4	8	2	6	6	3	5	9	7	9	10	6	10	6	8	8	5	10	1	6	5	6	4	2	9	5	7	6	10	8	6	3	7	7	6	3	6	5	7	8	9	5	4	2					
73	3	6	6	5	6	7	5	5	3	7	7	6	3	3	7	8	5	3	6	5	7	9	6	4	8	5	6	5	5	6	3	8	3	3	8	1	3	8	3	5	5	4	6	4	5	4	5	2	7	5	8	6	4	8	6	4	4	6	9	4	5	7				
74	4	8	4	8	3	6	6	3	5	4	5	3	5	6	8	5	8	6	3	6	2	4	10	5	6	8	4	8	3	6	4	8	10	5	4	5	5	5	4	7	2	4	9	8	1	10	5	6	7	8	5	9	8	5	5	4	7	5	4	6	6					
75	3	8	2	5	2	7	3	6	2	8	6	3	3	4	8	7	5	6	6	5	4	9	6	5	6	7	7	5	7	9	9	8	8	3	4	6	6	5	4	4	8	9	6	5	9	2	7	8	9	4	3	4	5	3	7	3	1	8	7	3	7	2				
76	3	3	8	9	3	5	5	8	6	5	6	7	1	7	4	5	8	8	7	6	2	3	6	7	5	6	5	4	4	3	5	6	10	4	5	5	5	4	7	3	7	6	2	7	9	5	4	6	6	4	4	5	6	5	9	9	6	5	4							
77	9	9	7	7	2	6	9	6	5	8	8	10	3	5	9	5	5	8	5	6	6	4	7	4	5	4	9	1	9	2	6	9	2	5	4	6	5	5	3	4	4	1	7	10	5	8	5	7	5	4	9	6	9	5	5	5	10	3	4	5	5	6				
78	6	3	5	6	8	4	7	8	9	2	3	4	5	9	4	5	6	5	4	5	2	7	5	6	7	2	6	3	2	4	4	4	2	5	5	8	2	5	5	5	5	8	8	8	8	6	7	9	4	4	3	6	1	4	5	10	5	4	6	7	5	2	9			
79	8	9	5	6	7	2	9	5	5	7	7	7	5	4	7	1	3	4	7	1	7	6	7	5	7	10	5	9	3	5	7	3	6	3	6	6	8	6	6	7	3	7	6	6	5	5	5	8	5	6	4	7	5	4	6	5	5	7	5							
80	8	3	7	2	8	5	8	9	4	7	7	5	3	7	4	6	6	5	5	6	5	4	6	7	1	6	4	9	8	3	7	3	6	5	7	7	2	5	2	5	2	5	2	5	7	4	1	5	5	4	2	6	7	6	4	6	9	7	8	4	6	6	5			
81	4	6	6	3	10	5	7	5	4	6	5	3	6	10	9	8	4	6	8	9	5	8	6	2	5	3	7	6	8	10	4	8	4	7	6	5	5	10	6	8	4	5	5	6	5	5	2	4	7	3	7	8	5	5	6	7	4	4	5	1	3	5				
82	7	2	6	7	6	8	9	7	6	2	3	6	3	2	4	1	4	6	2	2	5	6	4	5	8	7	6	8	7	3	5	6	6	4	5	7	5	9	3	5	6	8	3	5	7	5	7	7	5	2	3	8	7	2	7	9	5	6	7	6	8					
83	6	4	3	6	1	3	7	8	4	8	2	8	4	3	6	6	10	3	4	2	7	9	6	6	6	4	4	6	9	7	8	6	7	5	7	7	4	8	5	7	9	5	6	5	6	5	10	5	4	8	6	7	5	6	8	9	4	4	7	7	6	8	6			
84	7	4	4	6	6	7	7	7	4	8	2	6	5	9	4	3	5	8	7	5	4	4	2	6	7	2	8	8	9	6	8	5	8	6	5	4	2	4	6	5	4	6	8	5	7	8	3	5	6	5	8	5	6	7	4	3	10	7	10	4	4					
85	6	7	9	2	7	5	6	7	3	8	7	7	7	3	7	9	6	7	6	5	4	9	5	6	7	7	4	7	4	3	9	4	6	9	7	9	5	7	7	9	7	3	6	7	8	5	4	8	8	3	5	3	4	9	5	5	2	4	5	7	7	8	1			
86	2	9	8	5	7	4	6	8	5	3	7	6	9	8	7	4	7	4	5	7	9	2	6	5	4	3	6	4	7	7	7	7	8	4	4	6	8	3	4	6	9	4	5	4	8	8	2	7	7	8	9	5	1	5	4	8	9	2	4	2						
87	3	5	4	6	3	2	5	8	4	5	6	3	6	6	7	4	6	9	2	7	8	4	5	4	7	5	4	7	5	5	6	5	8	3	6	8	9	3	4	6	2	5	4	6	8	5	8	1	9	2	6	2	4	6	2	4	9	4	5	6	2	6	2			
88	6	6	4	6	7	6	8	7	5	8	6	6	3	7	1	4	5	6	4	1	6	5	6	6	2	5	2	2	8	6	2	2	4	2	2	9	3	2	6	3	6	5	1	5	5	6	5	6	5	3	3	7	4	4	3	3	4	6	5	4	5					
89	5	9	3	7	6	9	4	8	4	5	6	4	8	6	6	5	7	5	4	5	8	9	5	6	5	6	9	10	6	7	7	5	10	5	4	5	6	4	5	5	4	7	6	6	6	4	6	5	1	7	4	6	4	3	6	9	5	6	5	4	9	6	5	4		
90	7	2	6	9	4	6	2	4	5	6	6	7	5	9	5	5	7	5	4	6	2	6	8	7	6	8	1	6	4	4	5	3	8	8	3	6	8	4	4	5	6	8	5	7	8	5	4	5	8	4	6	3	6	8	7	5	9	3	2	6	4	7				
91	6	3	9	3	7	5	7	6	7	5	8	8	4	5	5	9	7	4	9	7	4	5	2	6	5	9	6	6	4	10	8	7	6	4	3	8	7	6	3	4	2	10	5	6	5	5	9	9	3	7	9	8	5	4	3	6	8	5	6	7	5	7				
92	8	4	5	5	8	3	3	2	7	6	6	8	8	8	7	9	8	6	5	5	4	4	5	2	5	4	5	5	1	3	6	3	6	6	6	5	6	5	1	6	7	6	3	6	4	4	5	7	6	5	10	5	1	6	6	2	8	4	6	7	3	4	8	6		
93	9	6	5	3	7	4	7	8	7	6	6	10	2	7	1	7	5	7	2	5	4	8	7	5	7	9	3	9	7	8	5	9	1	6	4	5	3	2	6	4	7	10	5	1	4	6	10	6	2	6	9	7	4	6	7	7	9	5	7	2	4	5				
94	4	6	2	5	5	6	7	6	4	5	4	8	5	4	4	6	8	9	7	5	8	6	6	9	6	6	2	5	5	5	5	3	4	4	1	8	1	4	5	8	7	3	7	6	7	6																				

Set Data DUN62
Ukuran Mesin

Nomor Mesin	Panjang	Lebar	Nomor Mesin	Panjang	Lebar
1	15	14	32	17	8
2	13	10	33	21	19
3	16	14	34	21	10
4	20	13	35	15	10
5	16	13	36	12	9
6	21	14	37	21	17
7	19	17	38	16	9
8	19	14	39	18	14
9	21	21	40	15	9
10	20	17	41	17	12
11	13	11	42	17	12
12	14	12	43	11	9
13	19	18	44	20	8
14	21	17	45	21	17
15	21	20	46	20	13
16	21	7	47	19	10
17	20	19	48	20	14
18	16	9	49	18	10
19	17	11	50	13	8
20	16	15	51	18	11
21	14	10	52	16	10
22	19	16	53	20	10
23	20	15	54	19	19
24	18	9	55	21	11
25	18	14	56	20	7
26	14	14	57	11	7
27	16	11	58	17	11
28	12	12	59	21	11
29	17	13	60	13	7
30	13	10	61	21	10
31	14	13	62	17	16

Simulated Annealing

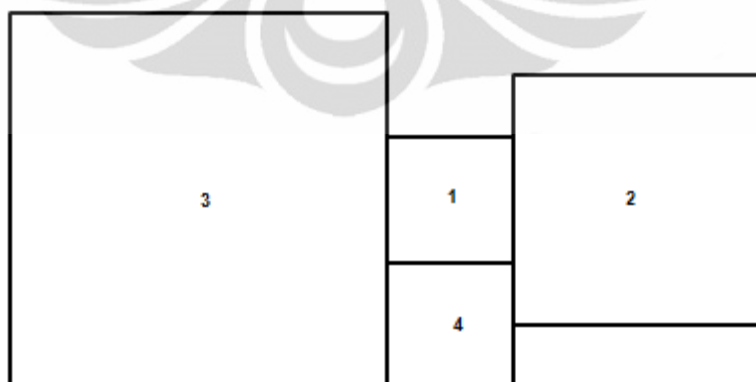
KH1



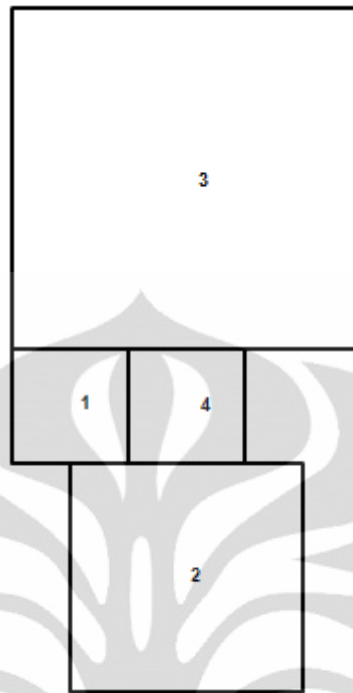
KH 2



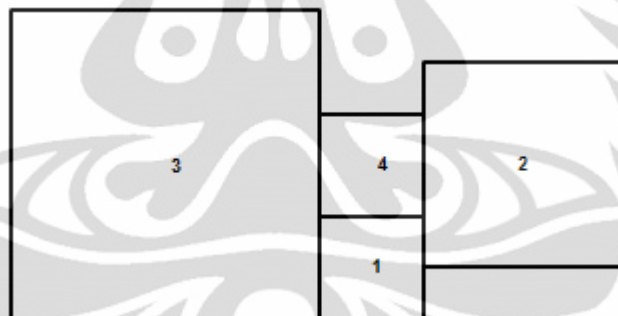
KH 3



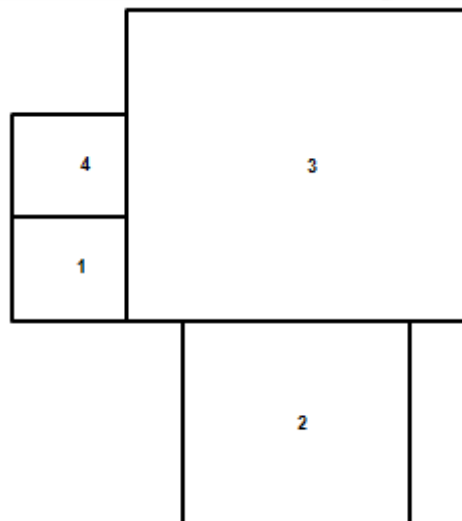
KH 4



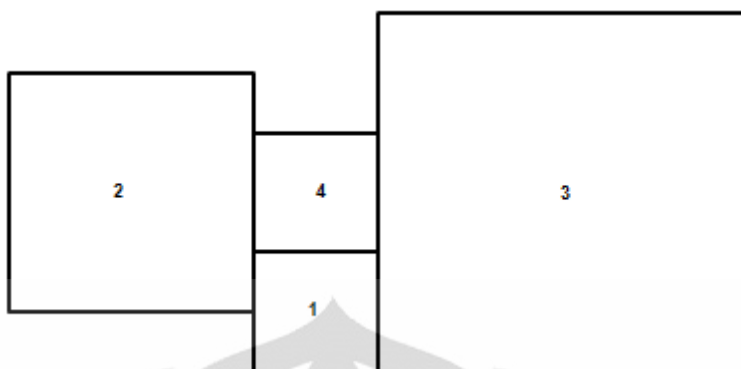
KH 5



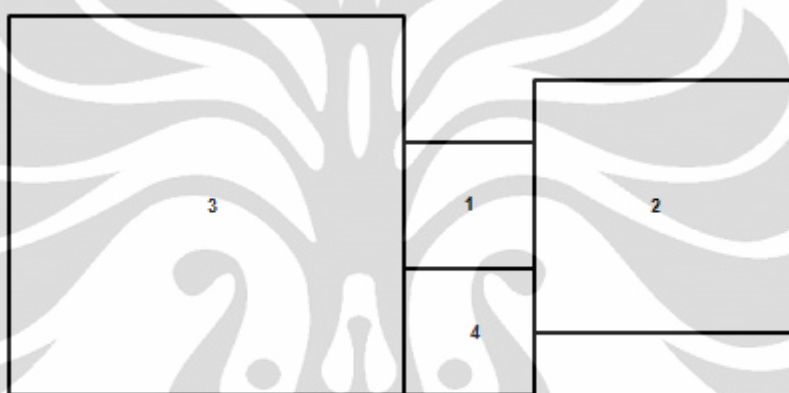
KH 6



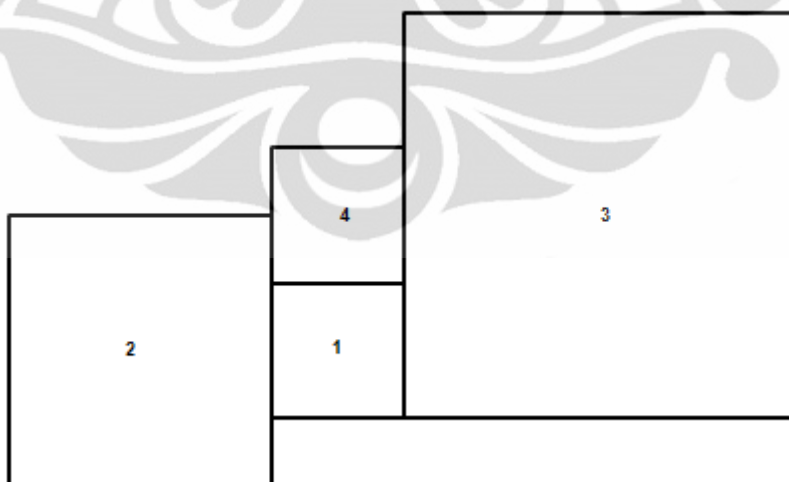
KH 7



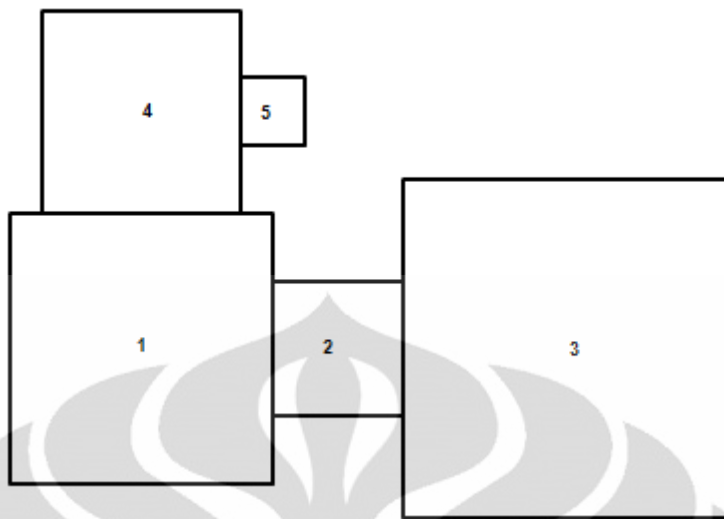
KH 8



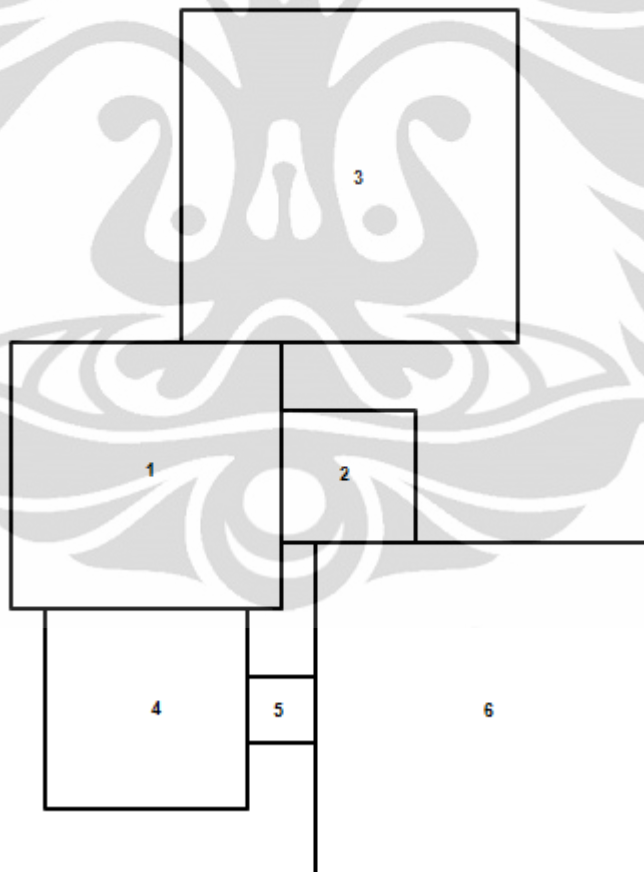
KH 9



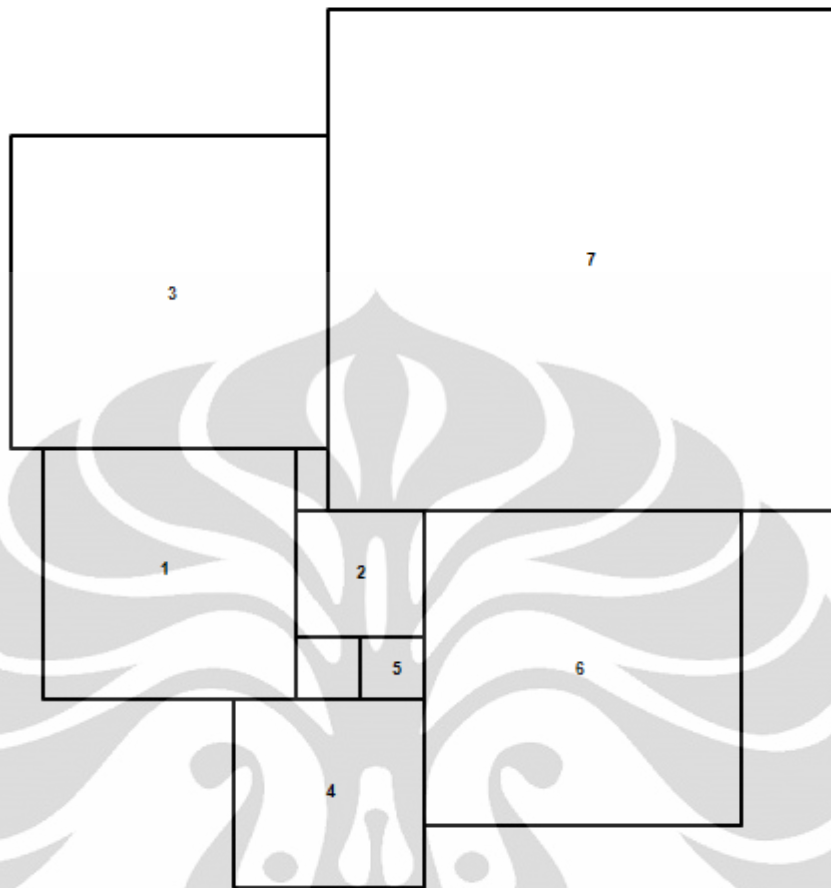
NUG 5



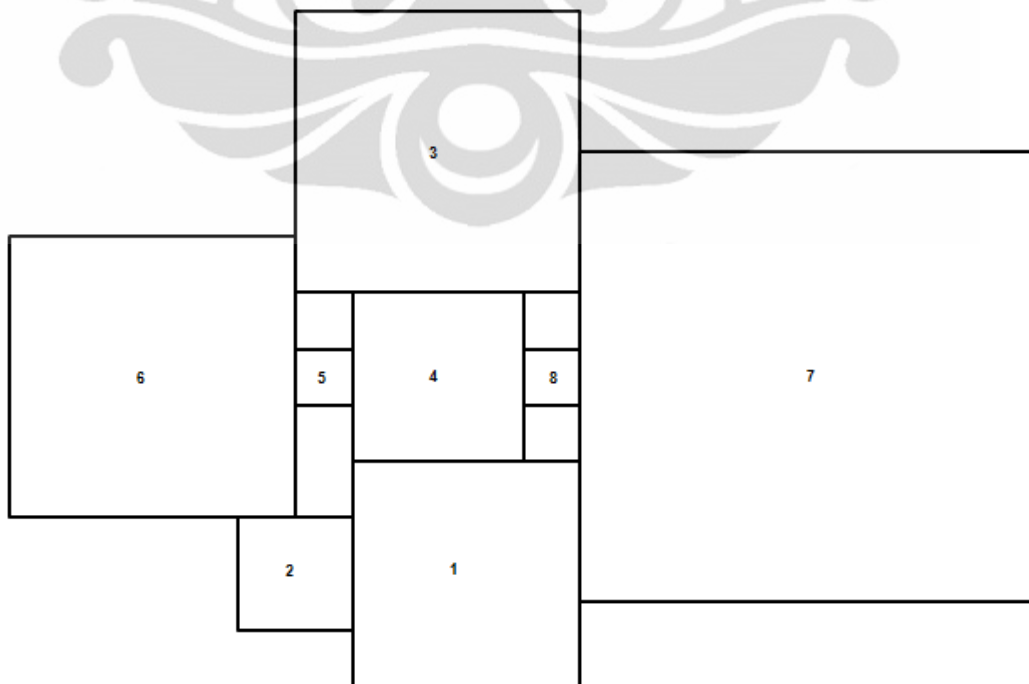
NUG 6



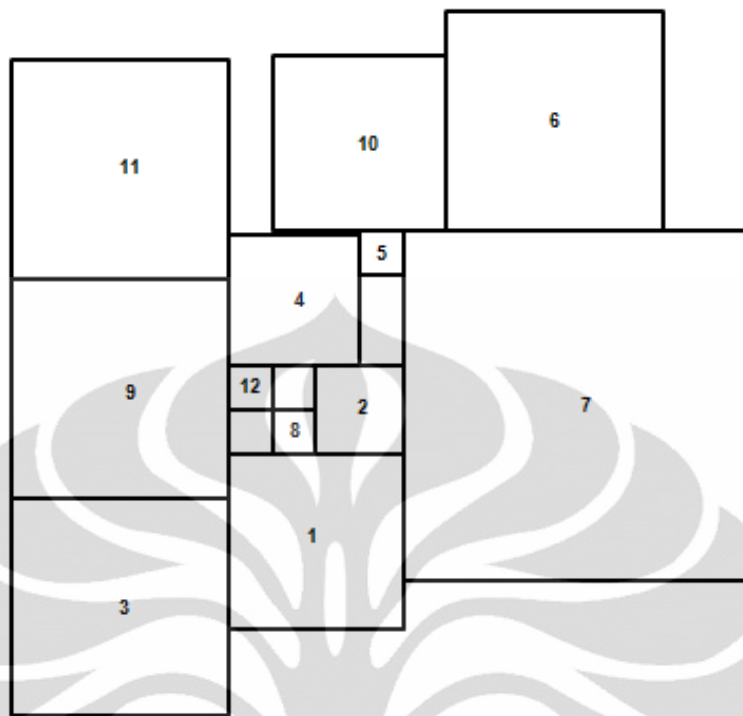
NUG 7



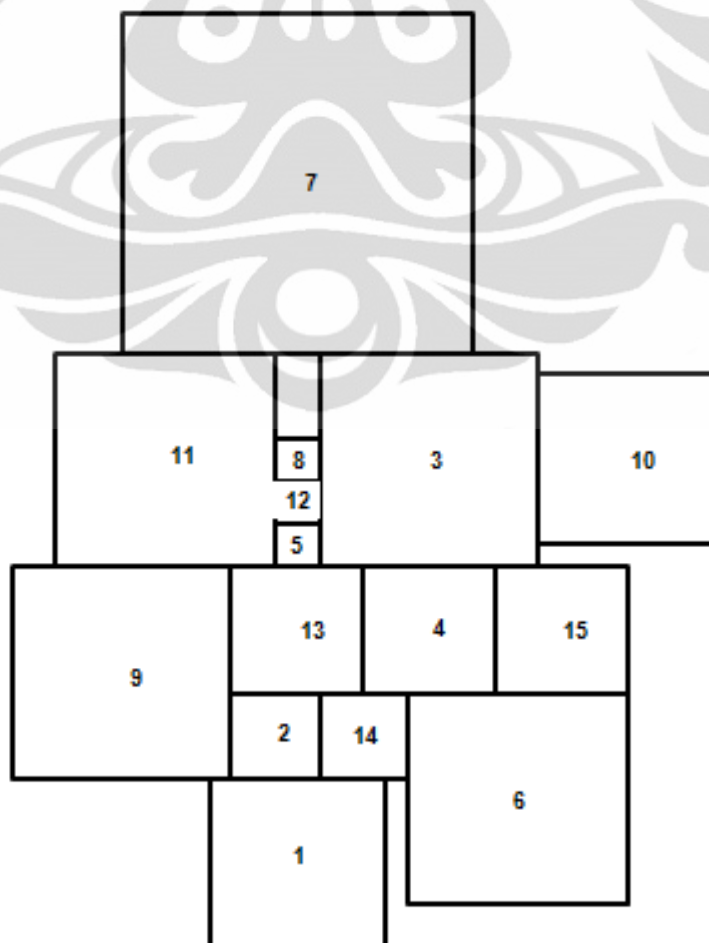
NUG 8



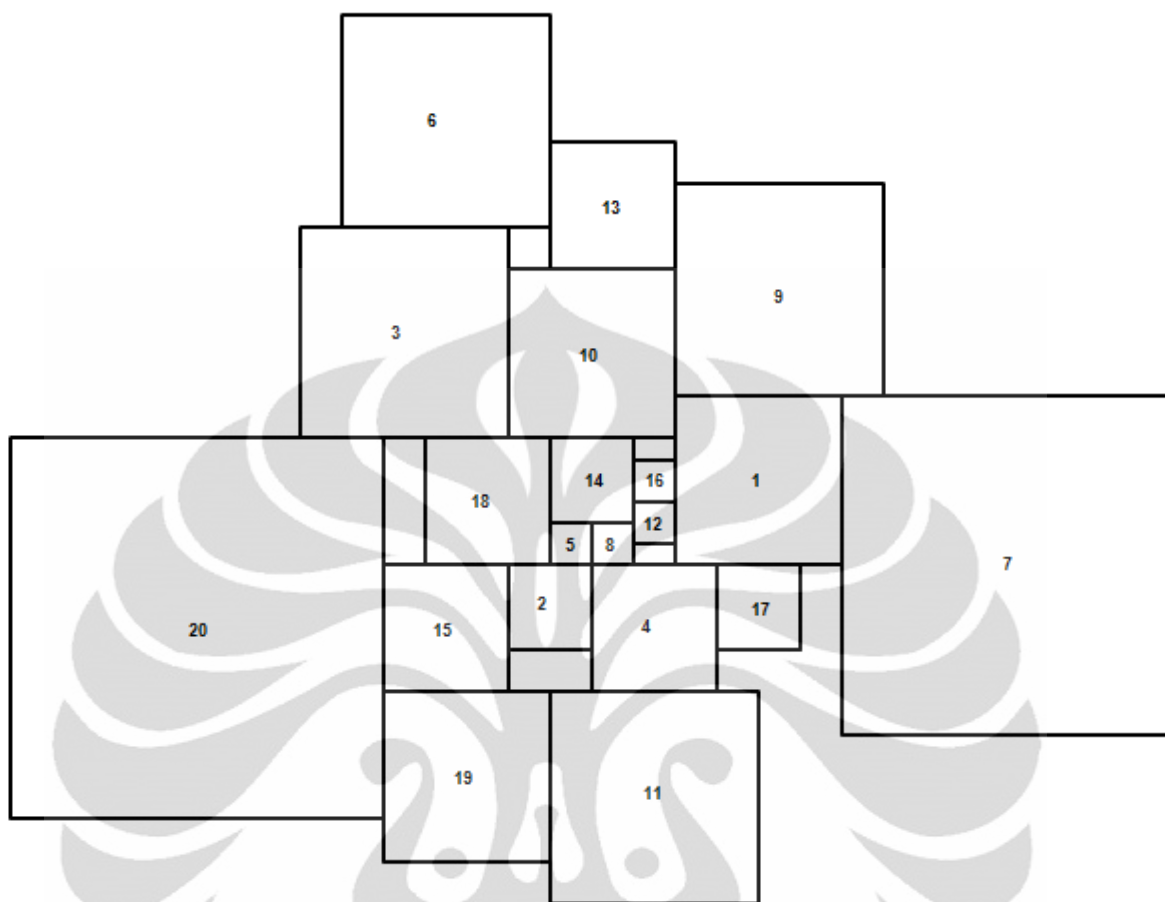
NUG 12



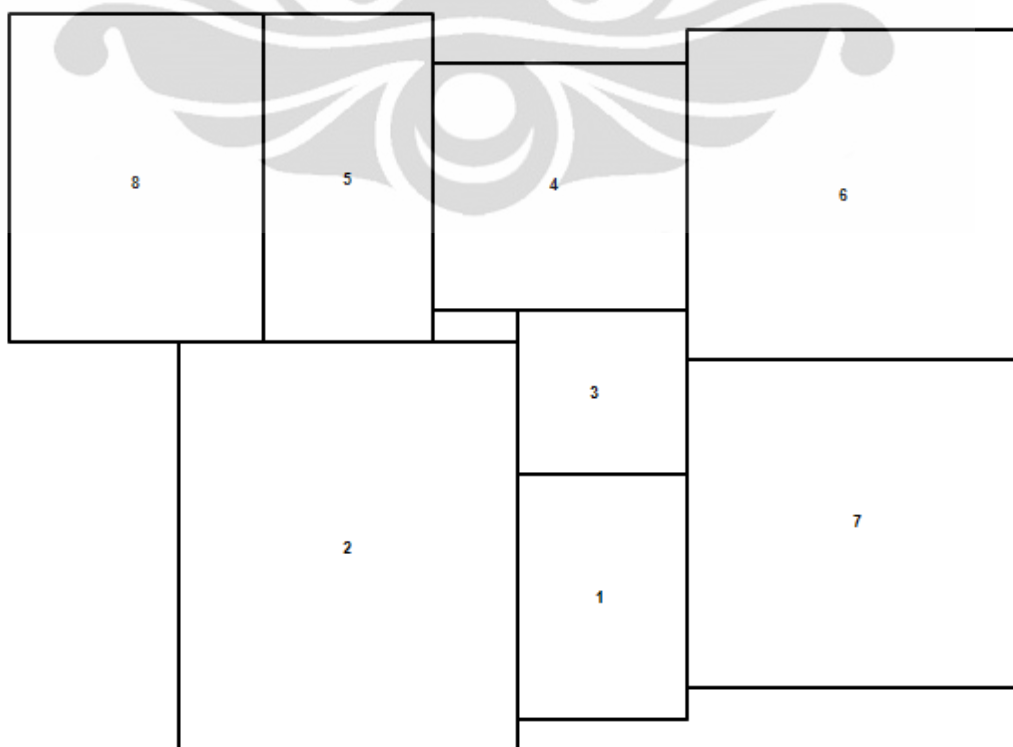
NUG 15



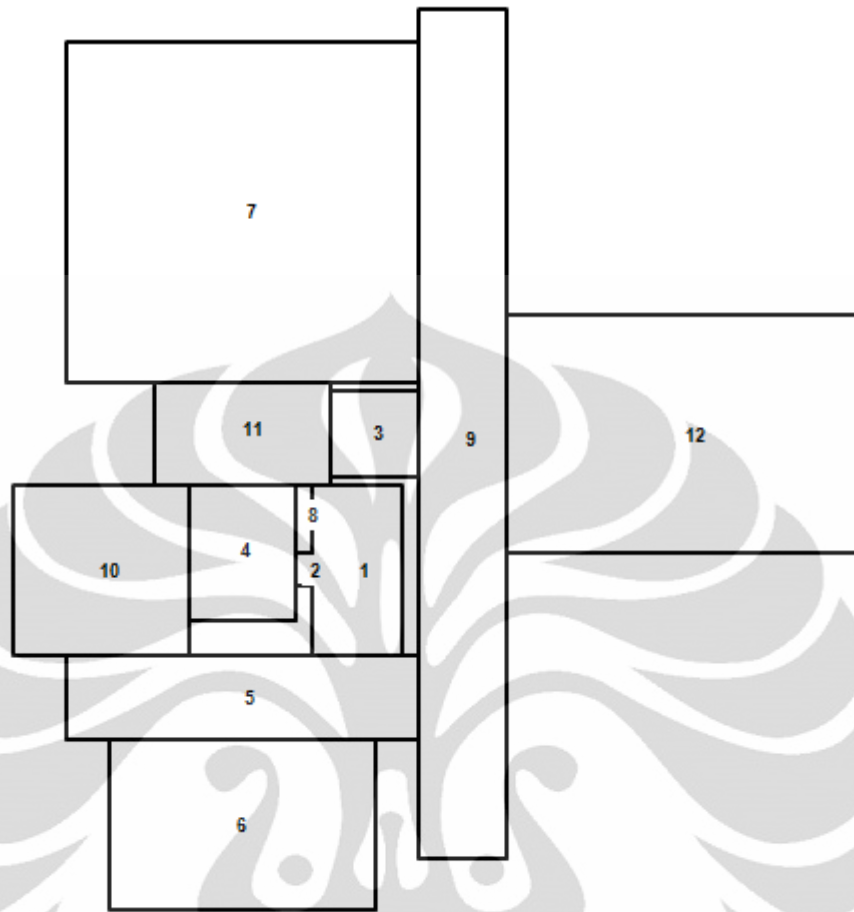
NUG 20



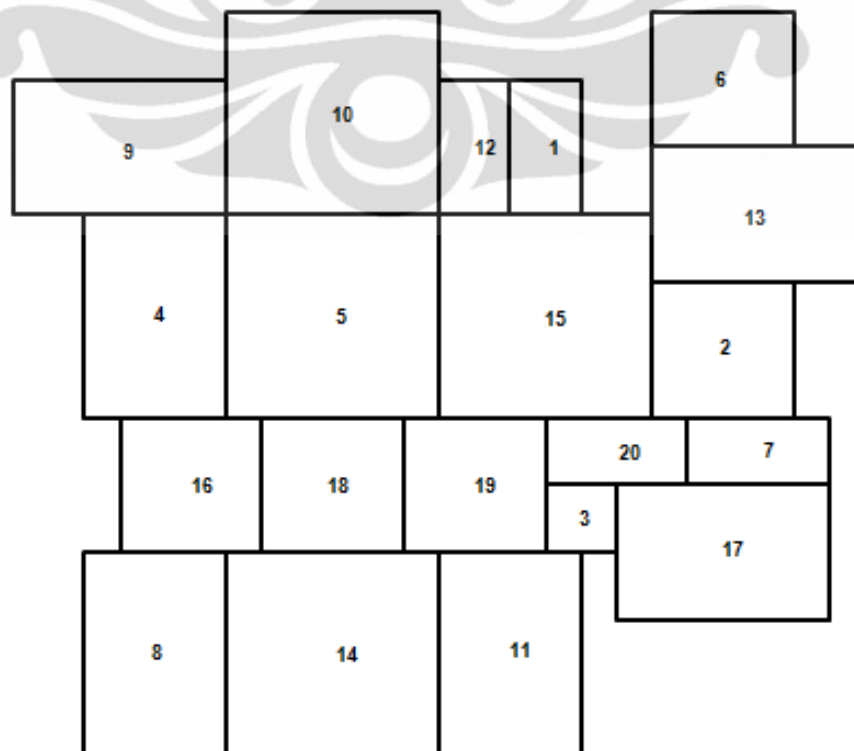
OPT 8



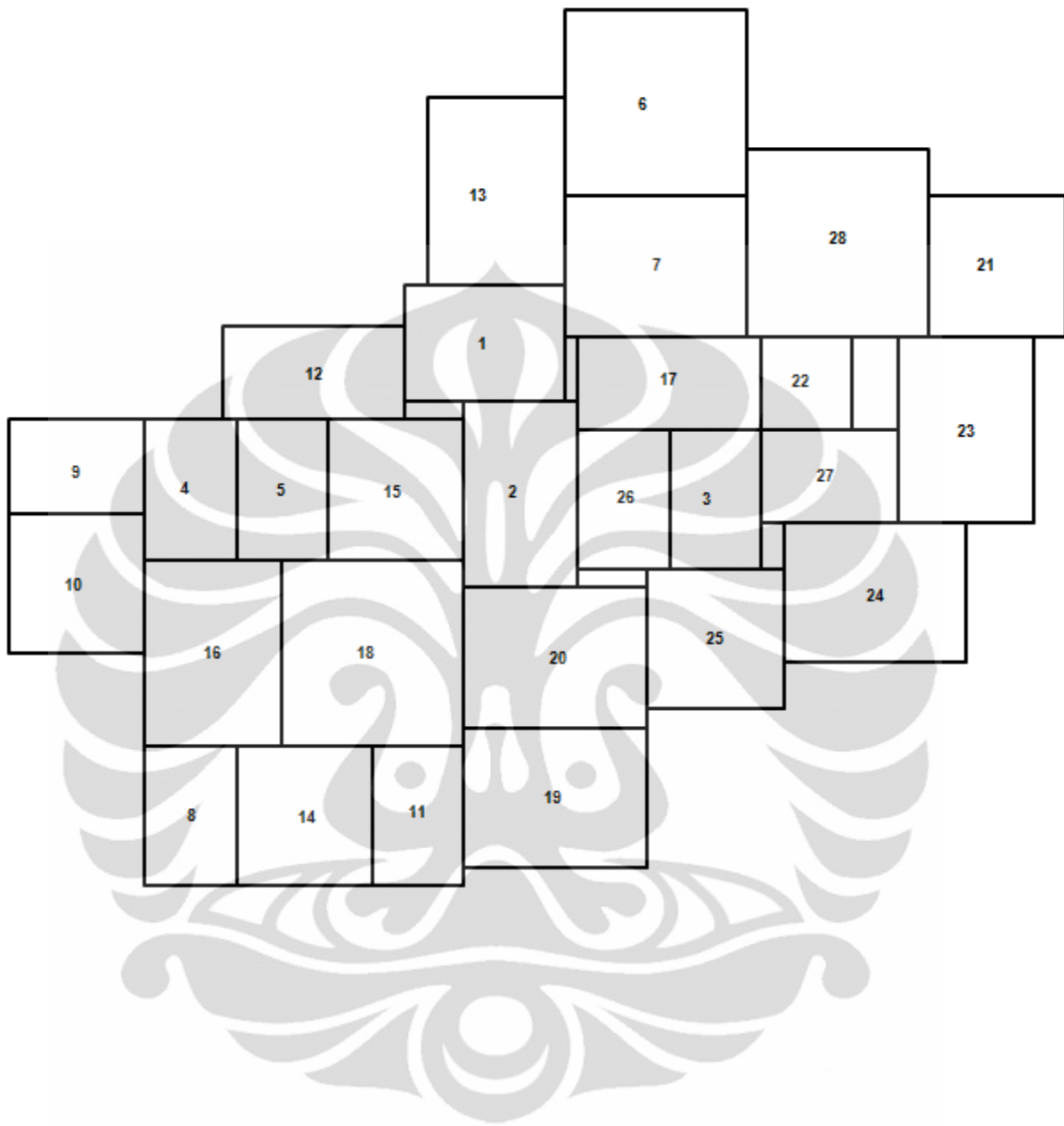
OPT 12



OPT 20



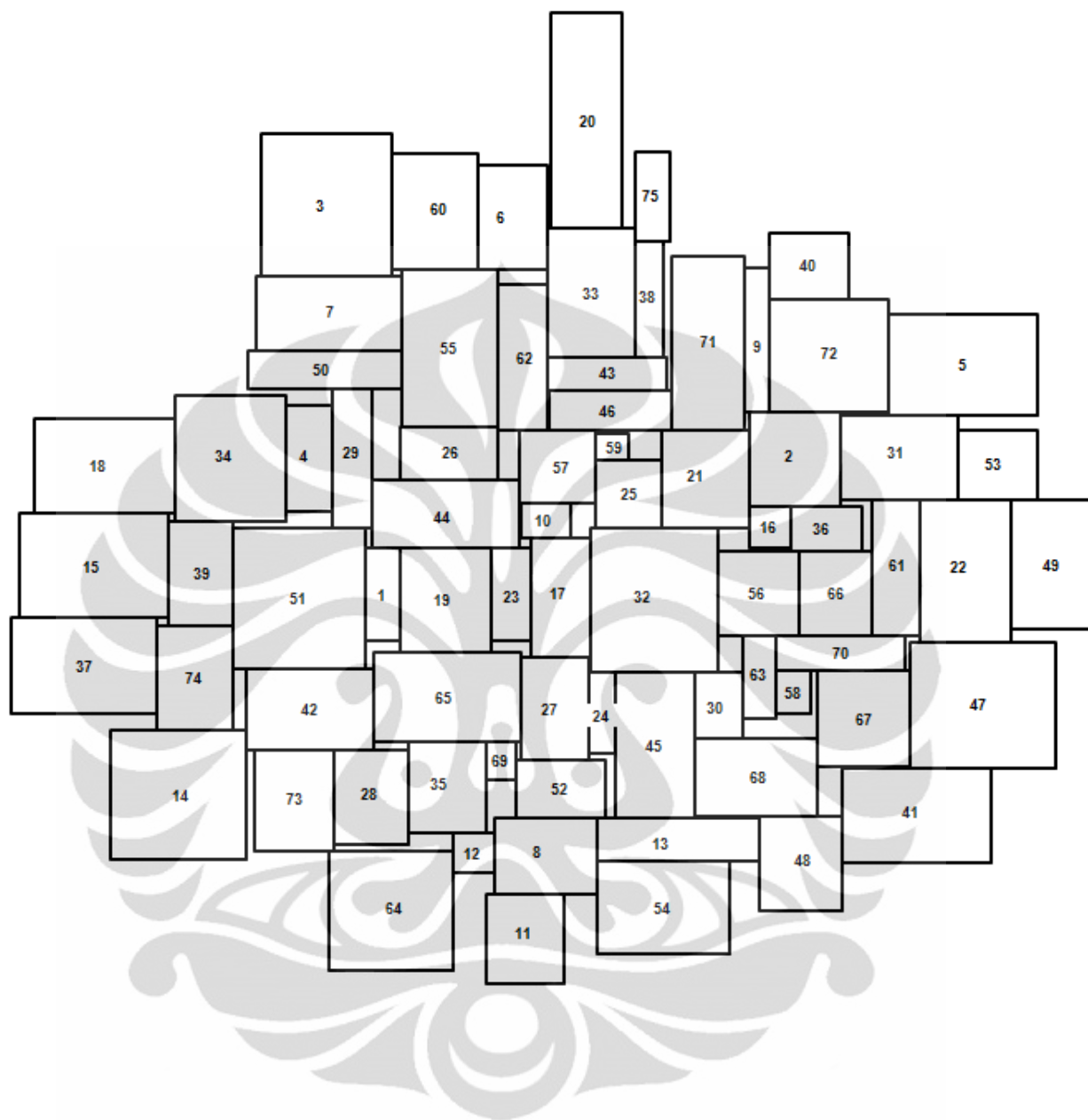
OPT 28



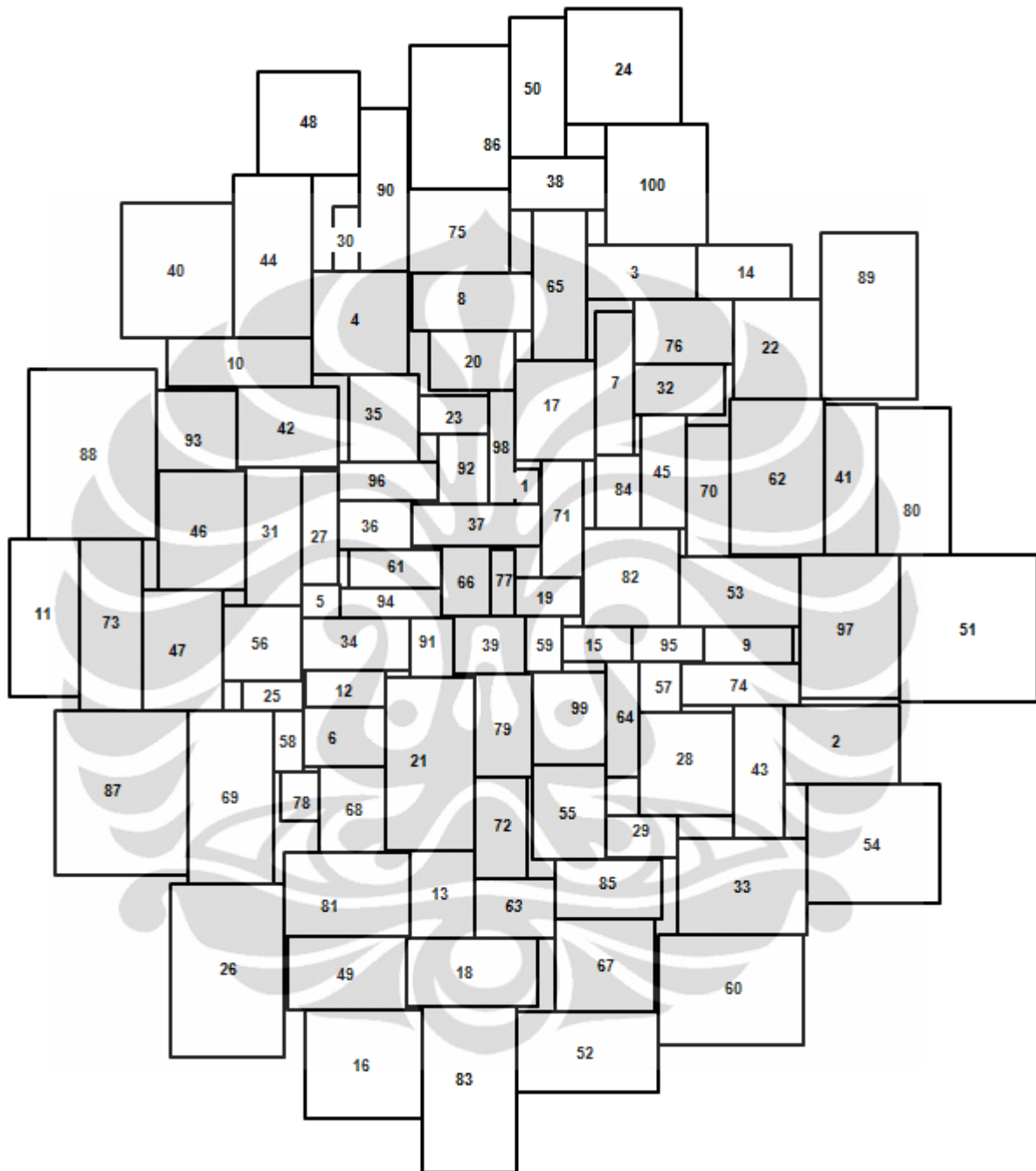
OPT 50



OPT 75



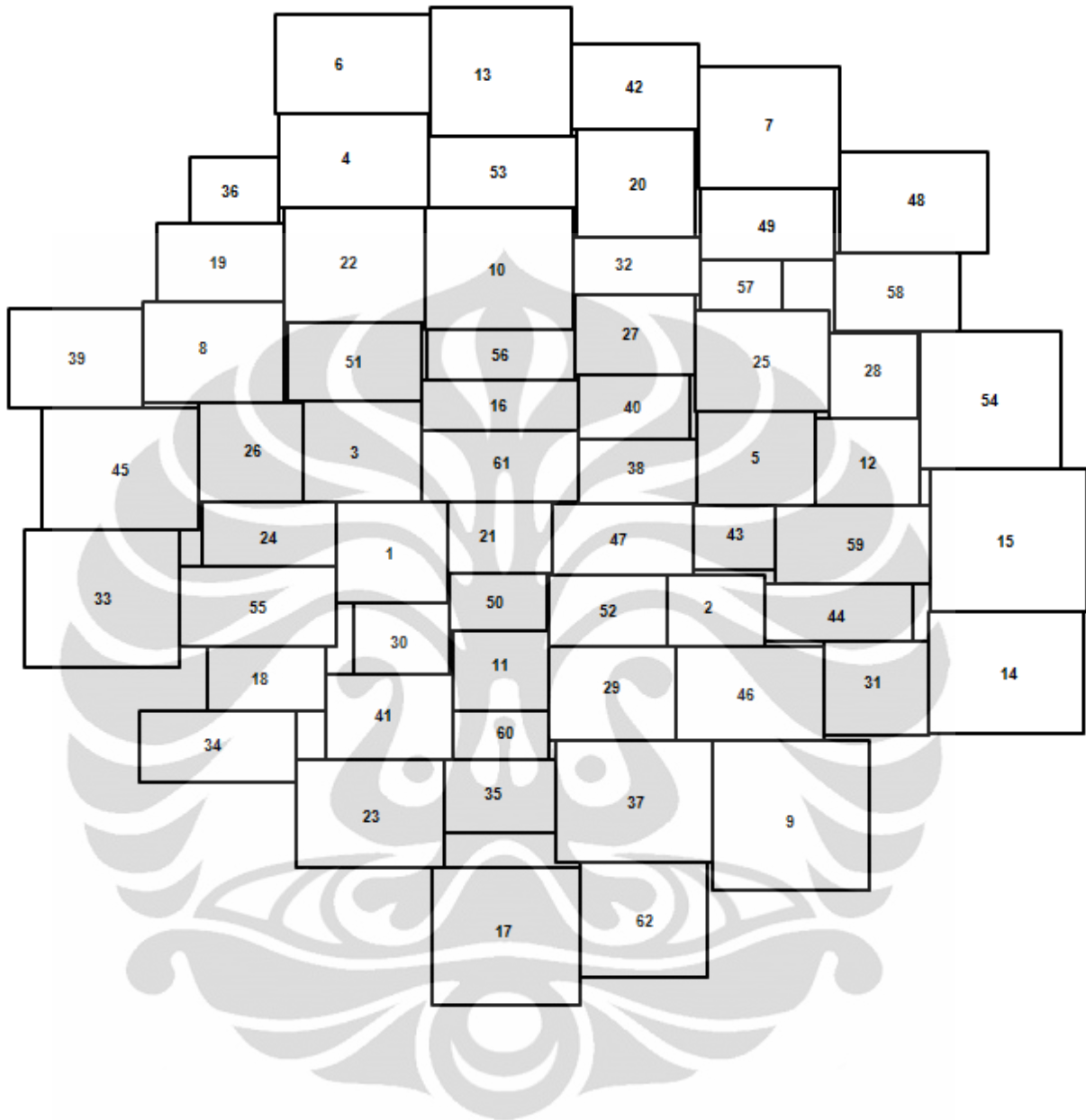
OPT 100



OPT 125

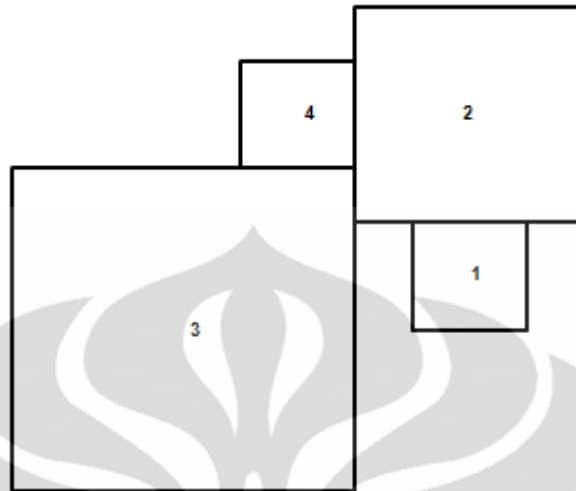


DUN 62

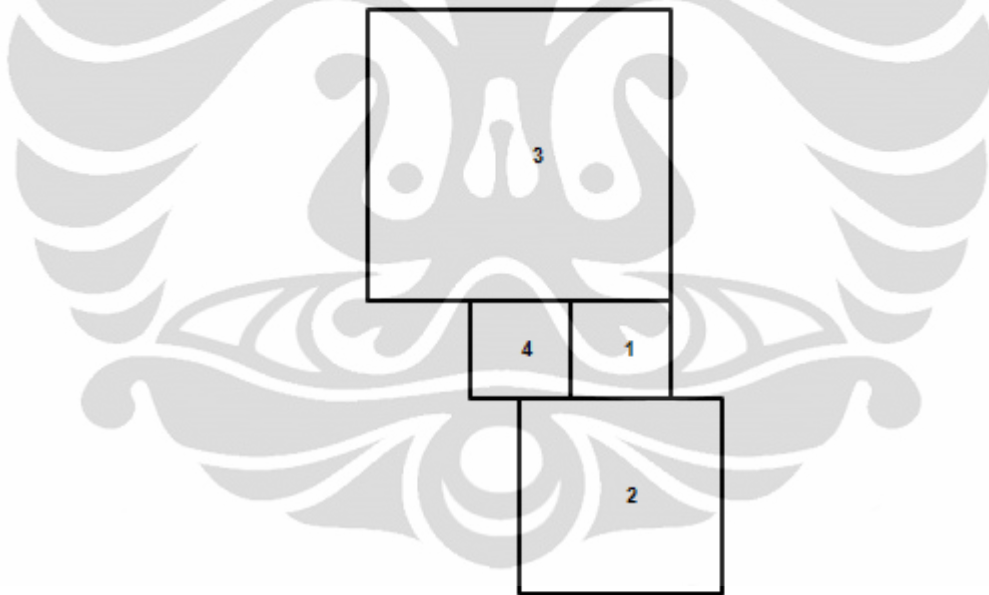


Tabu Search

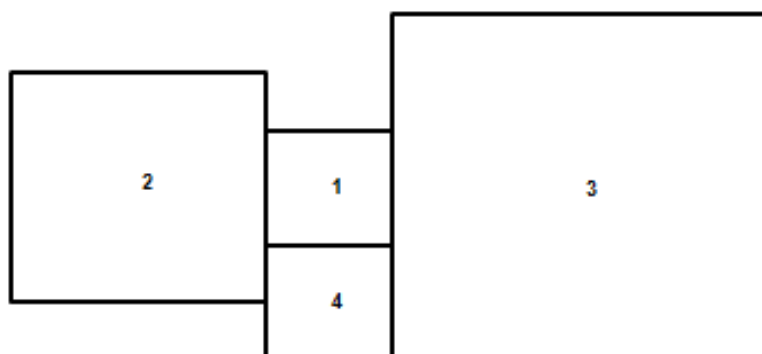
KH1



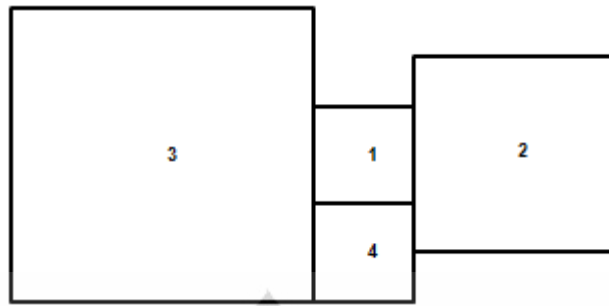
KH 2



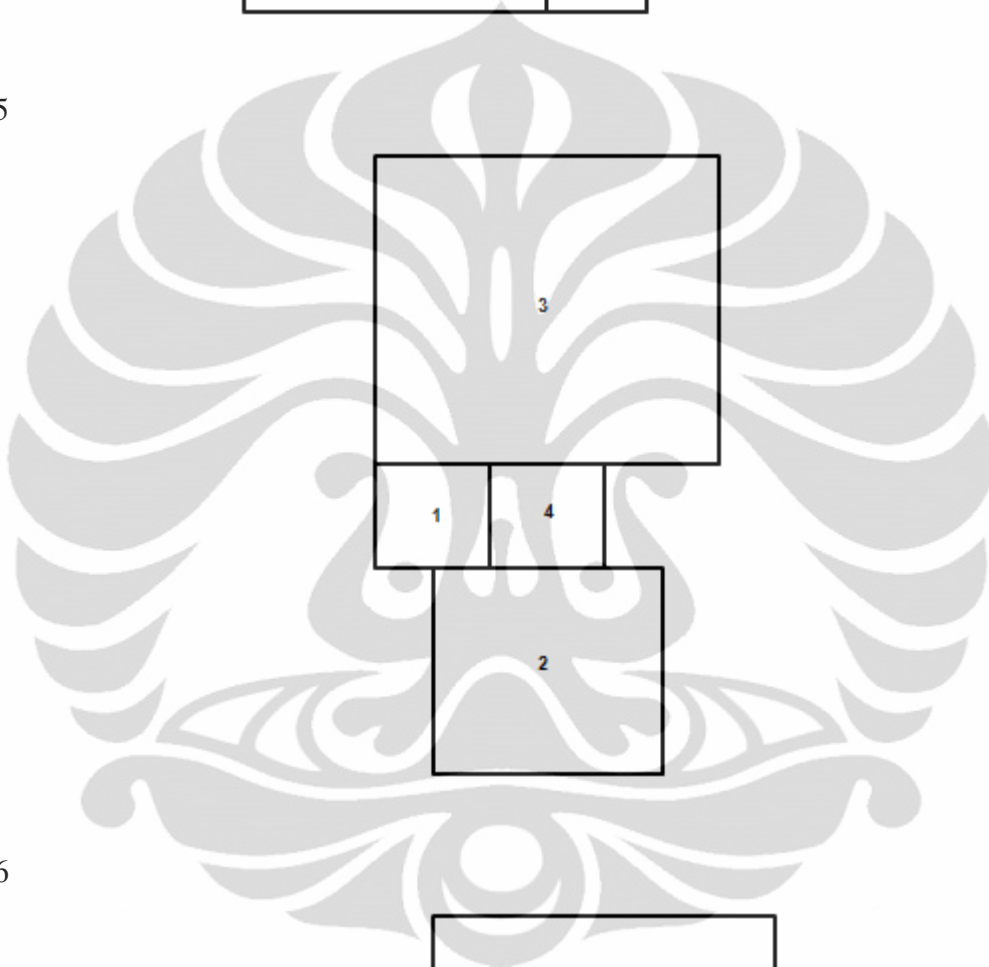
KH 3



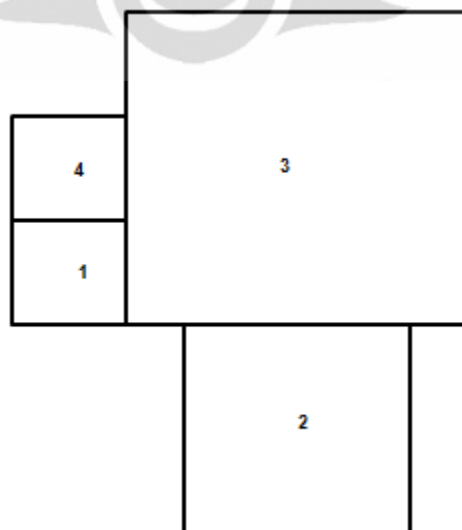
KH 4



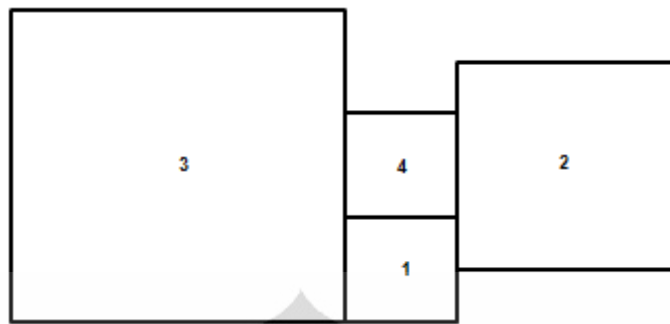
KH 5



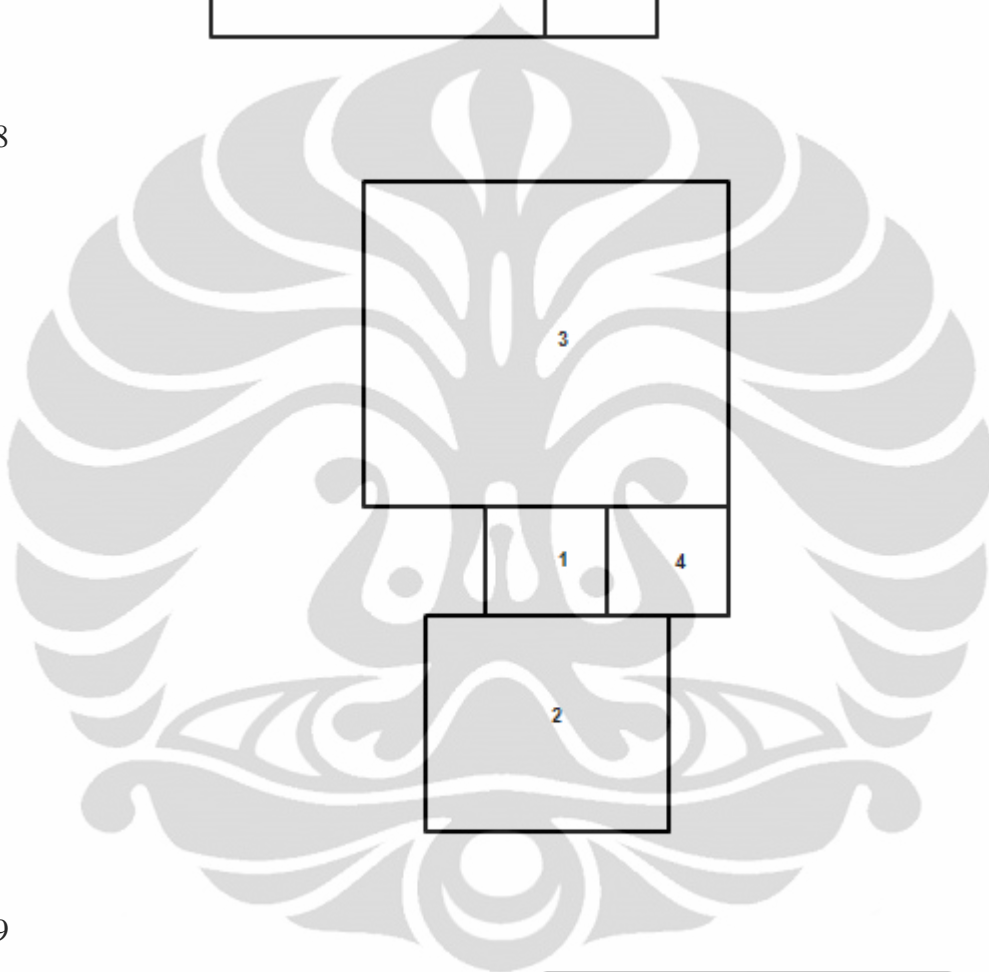
KH 6



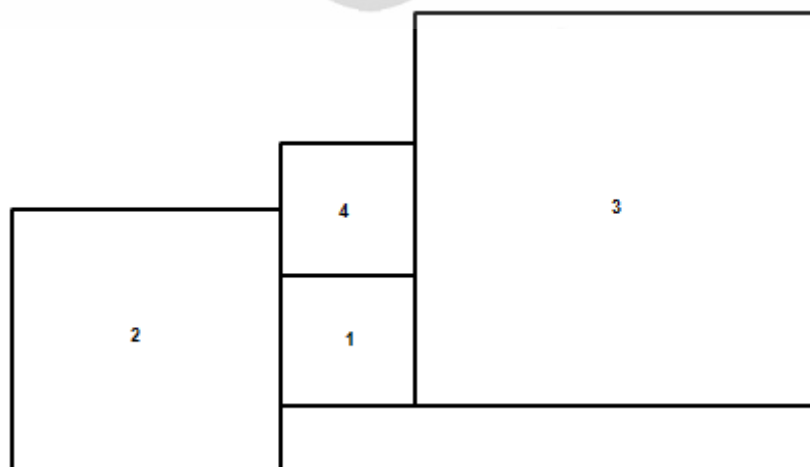
KH 7



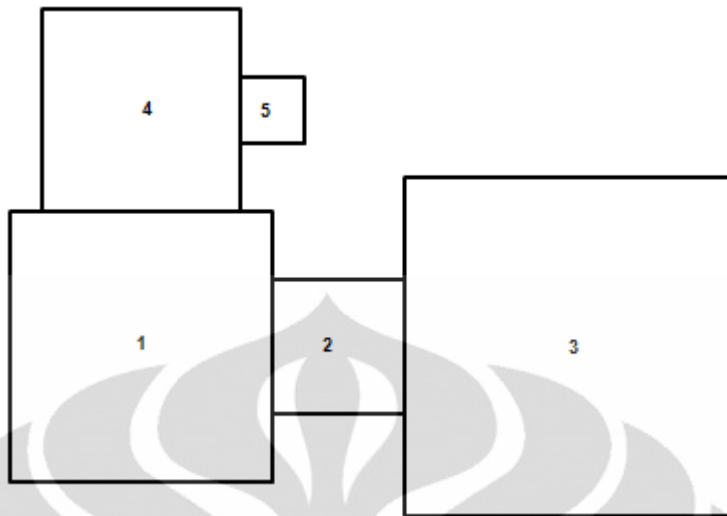
KH 8



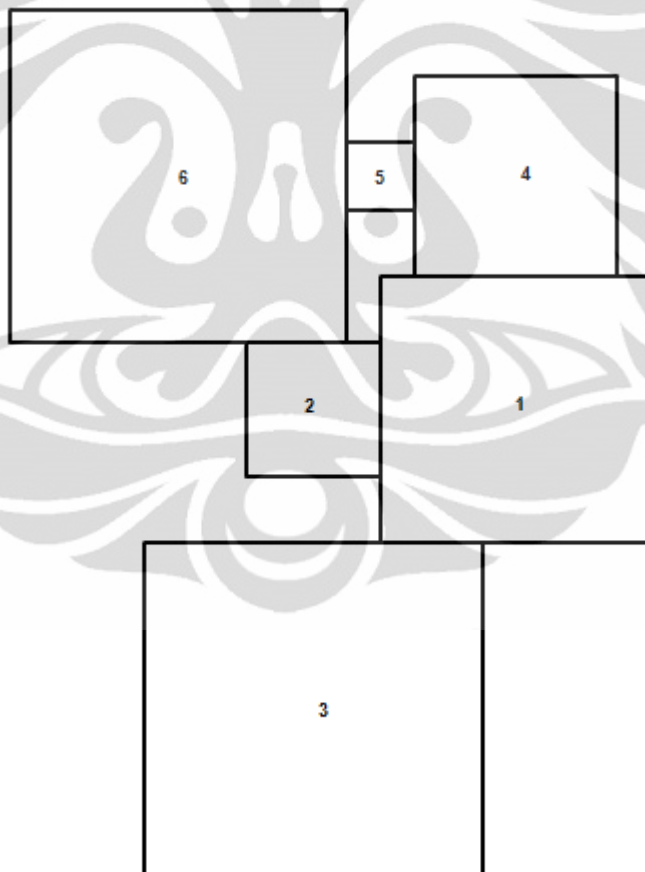
KH 9



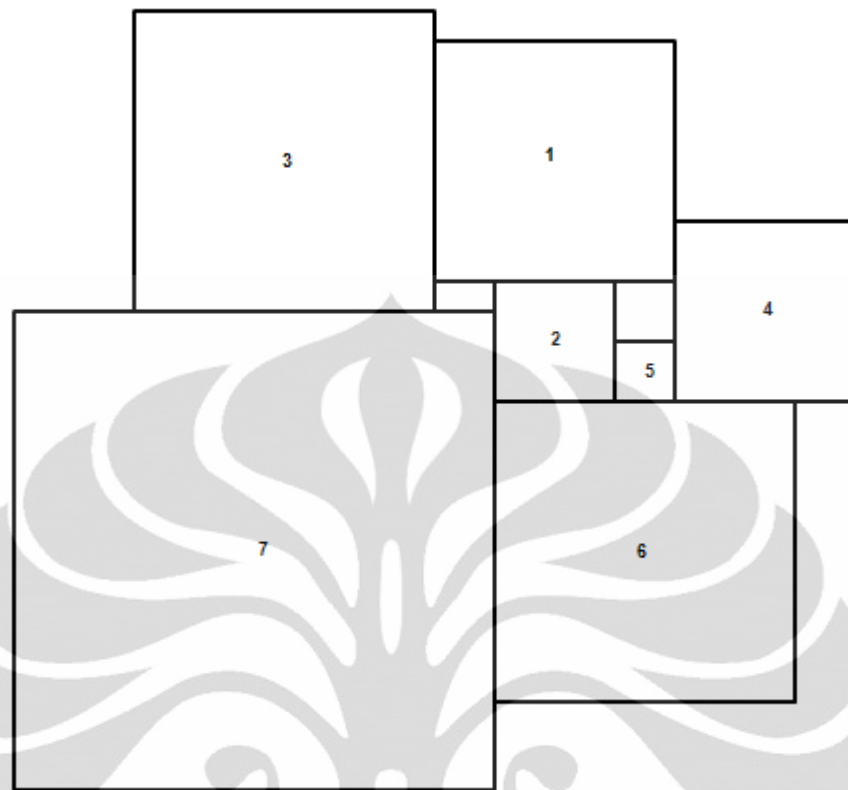
NUG 5



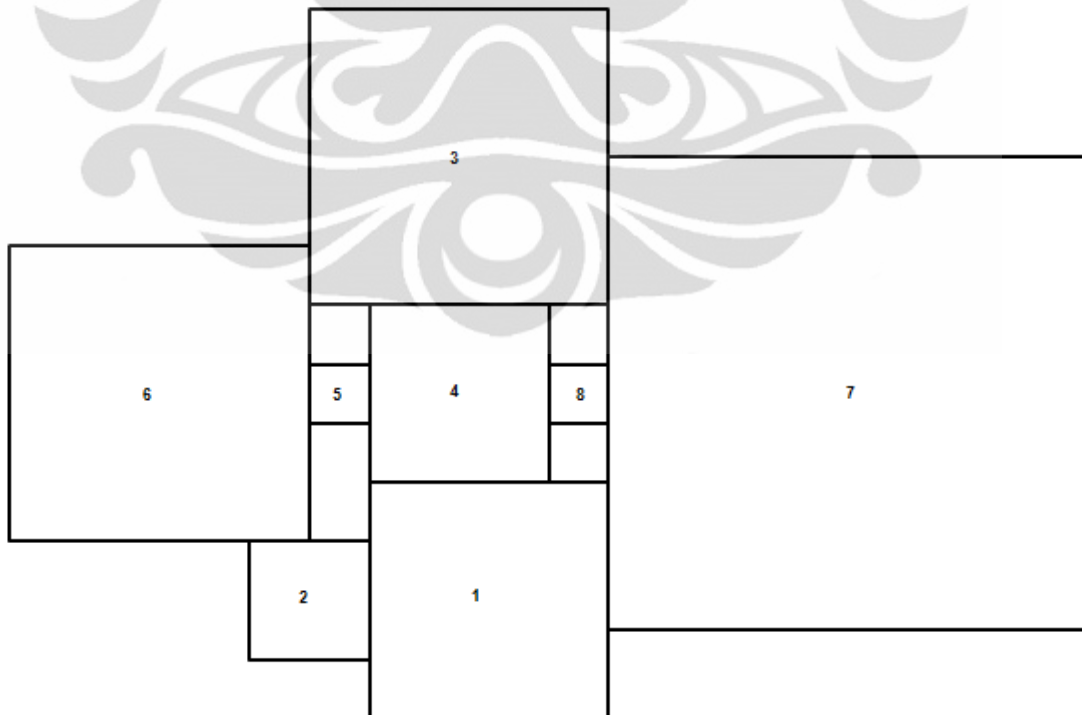
NUG 6



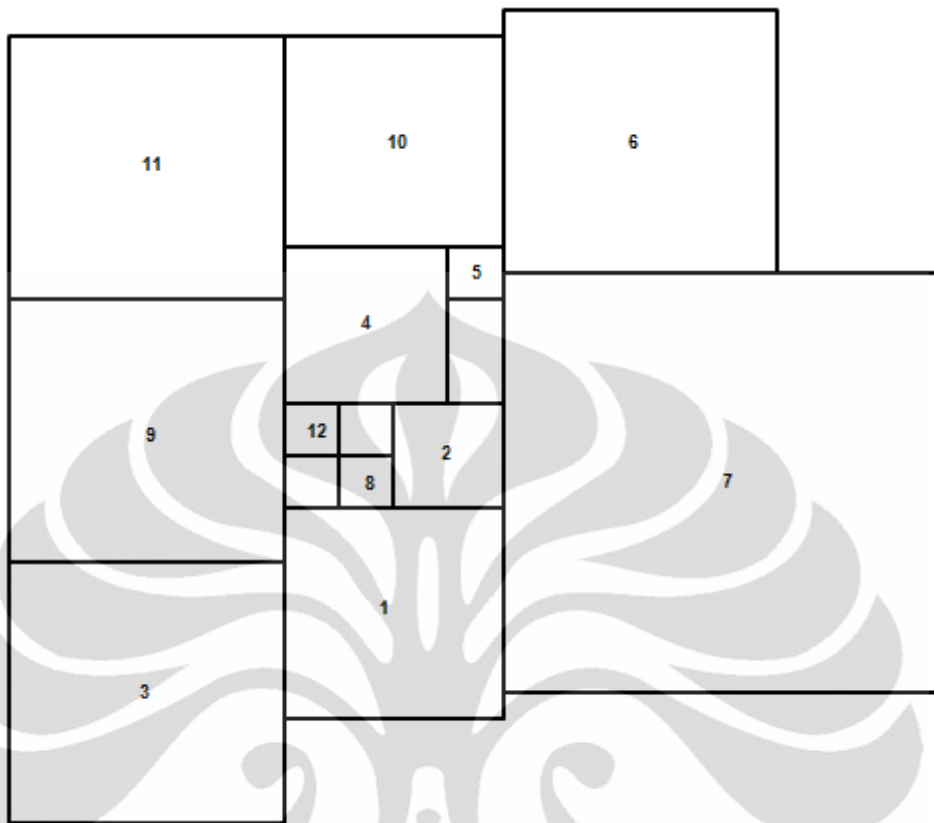
NUG 7



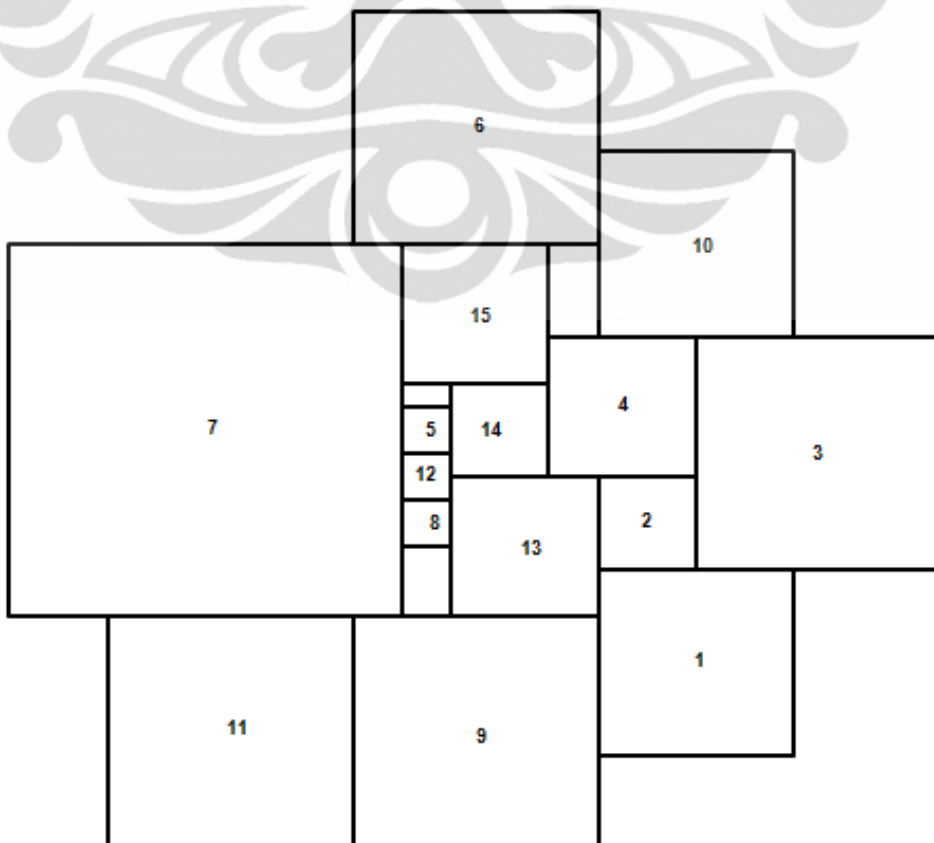
NUG 8



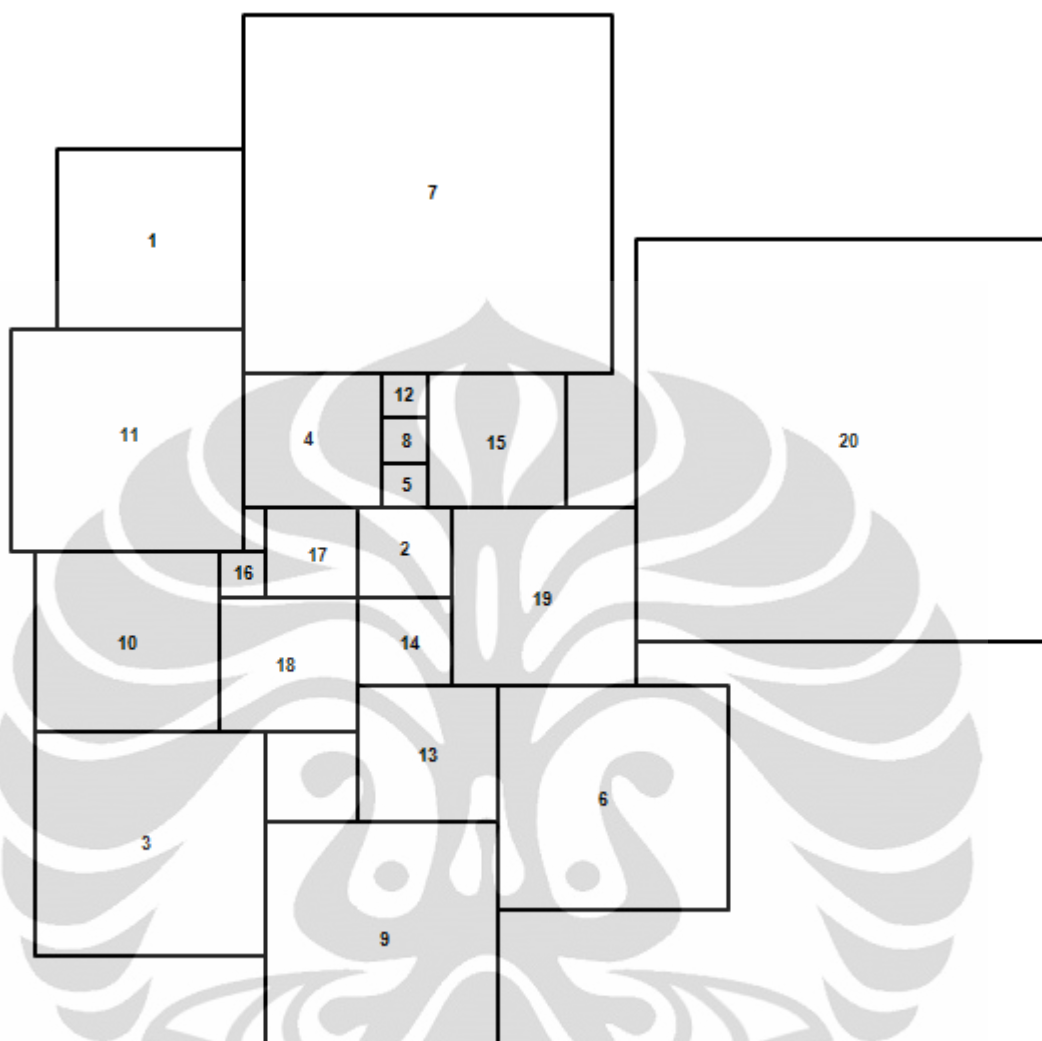
NUG 12



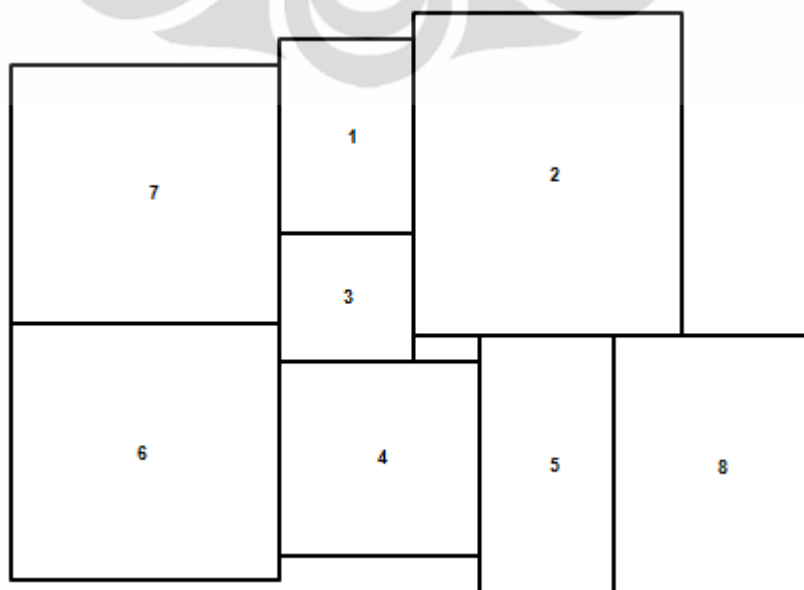
NUG 15



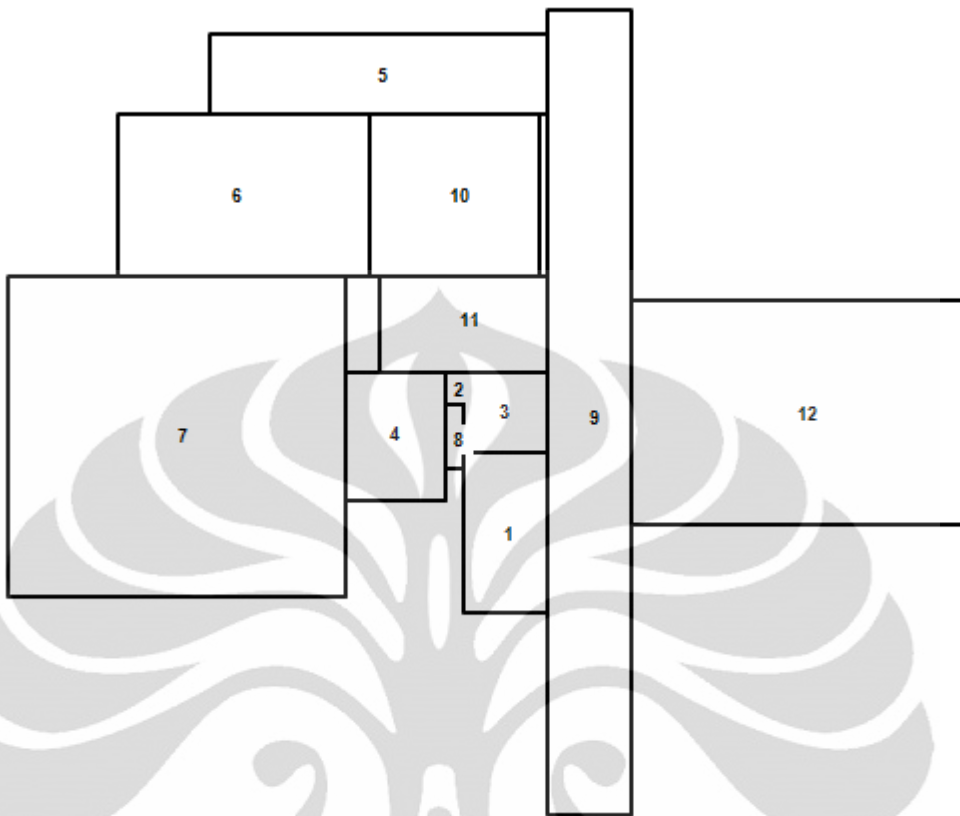
NUG 20



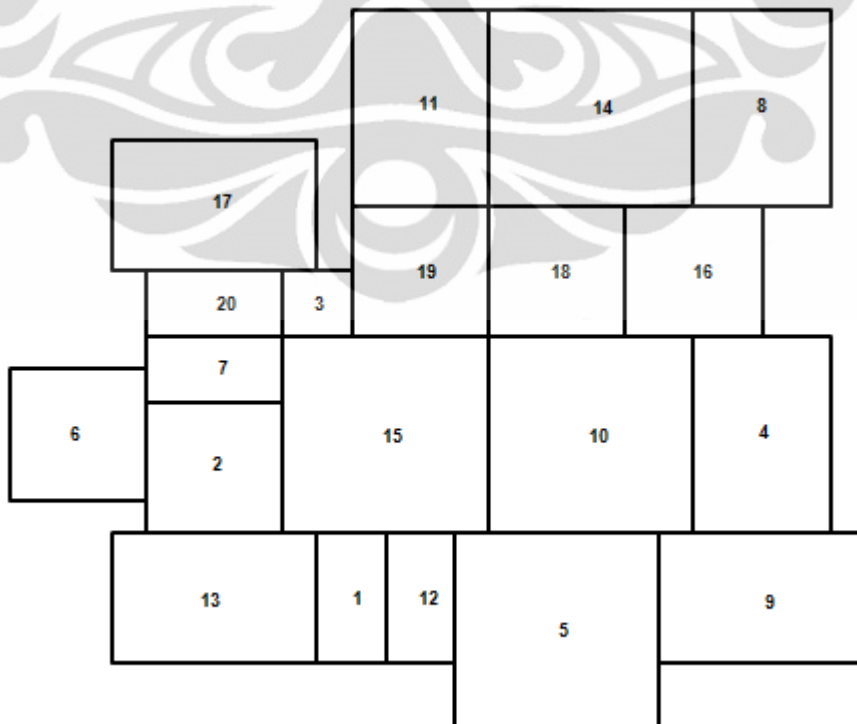
OPT 8



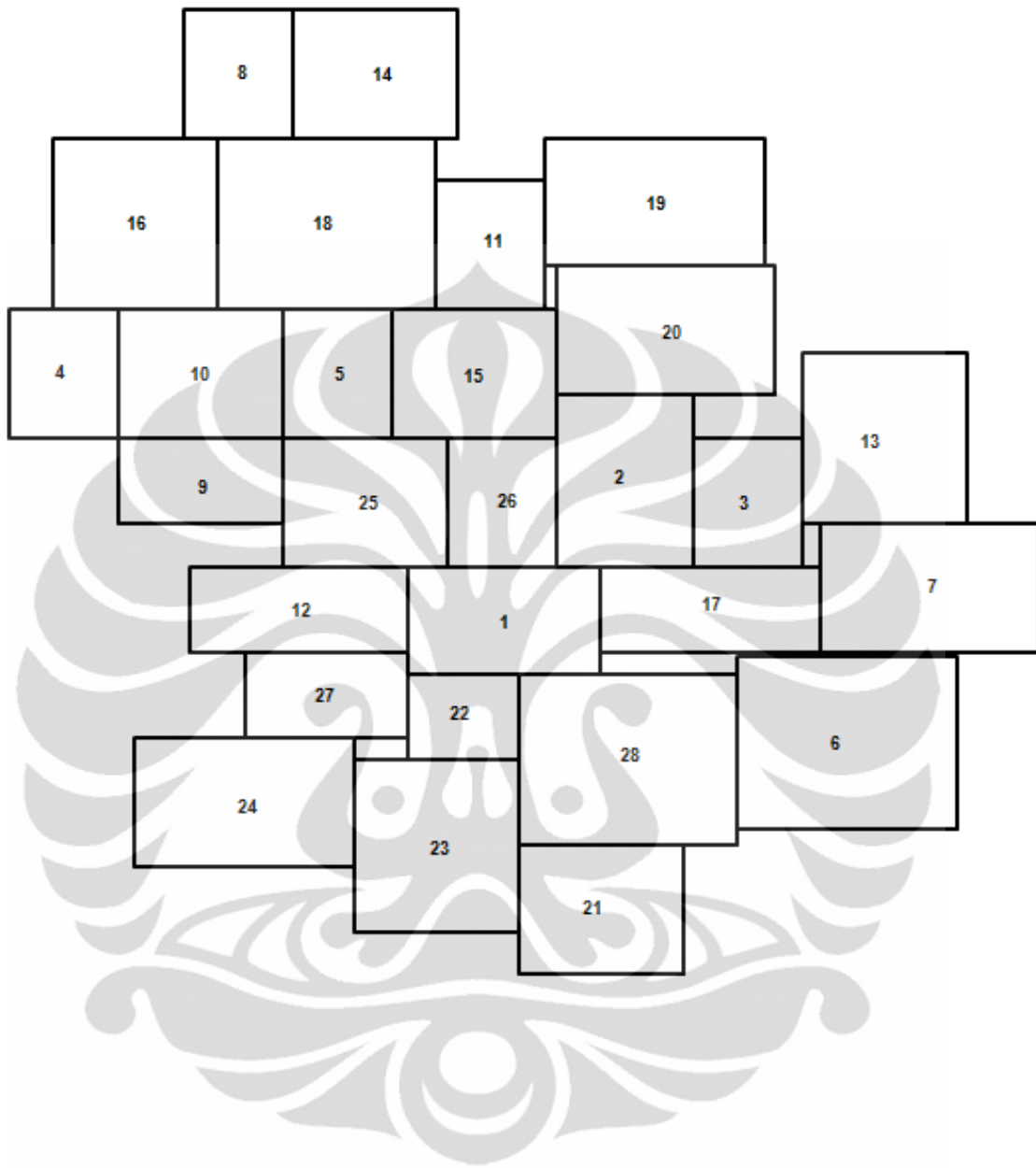
OPT 12



OPT 20



OPT 28



OPT 50



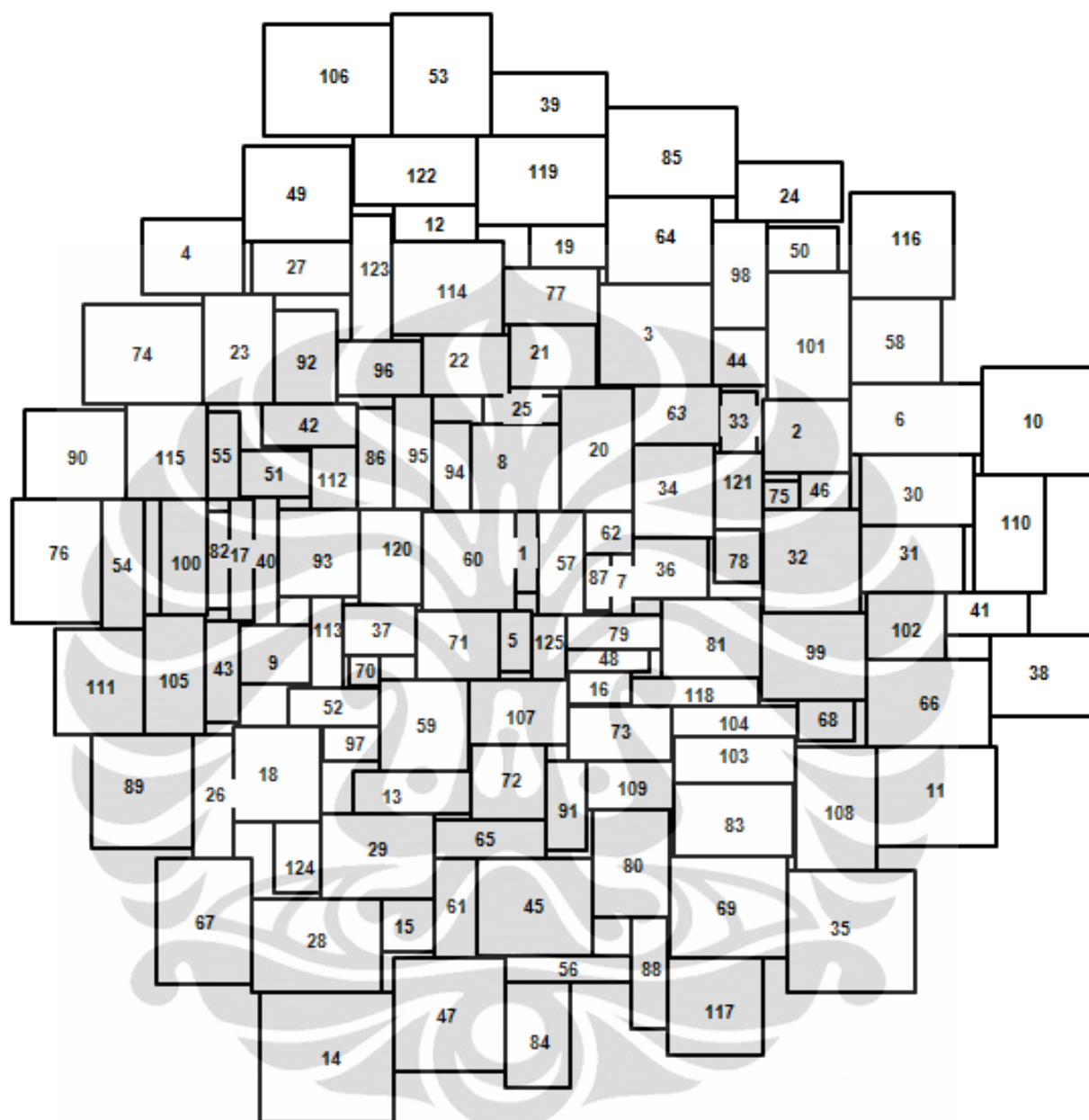
OPT 75



OPT 100



OPT 125



DUN 62

