**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**INSTITUT TEKNLOGI SEPULUH NOPEMBER**

**USULAN TUGAS AKHIR**

# IDENTITAS PENGUSUL

**NAMA : NABRIS ZALMI PRATAMA**

**NRP : 5109100062**

**DOSEN WALI : Ary Mazharuddin S., S.Kom., M.Comp.Sc.**

**DOSEN PEMBIMBING : 1. Victor Hariadi, S.Si., M.Kom.**

**2. Rully Soelaiman S.Kom., M.Kom.**

# JUDUL TUGAS AKHIR

“Implementasi *Integer Linear Programming* dengan Metode *Heuristik* untuk Optimasi Pengemasan Peti Kemas”

“*Container Loading Optimization with Integer Linear Programming and Heuristic*”

# LATAR BELAKANG

Peti kemas merupakan sarana kegiatan pengiriman barang dari satu tempat ke tempat lain. Pada proses pengemasan barang ke peti kemas terdapat batasan ukuran peti kemas dan biaya pengemasan. Setiap industri pasti menginginkan pengiriman barang yang efektif dan efisien untuk mendapatkan keuntungan yang maksimal. Agar mendapatkan keuntungan yang maksimal, biaya pengemasan harus seminimal mungkin tetapi memenuhi permintaan yang ada. Oleh karena itu, diperlukan suatu strategi pemecahan masalah yang bisa memberikan solusi optimal.

Terdapat dua langkah dalam menyelesaikan permasalahan pengemasan ini. Langkah pertama, menentukan pola pengemasan barang yang baik dan efektif di dalam peti kemas. Langkah kedua, menggunakan *Integer Linear Programming* (ILP) untuk mendapatkan solusi optimal berupa biaya proses pengemasan yang minimal.

Tujuan utama dalam permasalahan ini adalah untuk mendapatkan biaya proses pengemasan yang minimal dengan pengemasan barang ke peti kemas yang baik dan efektif.

# RUMUSAN MASALAH

Adapun rumusan masalah untuk tugas akhir ini adalah :

* + - 1. Bagaimana konsep penerapan pengemasan barang ke peti kemas yang baik dan efektif?
      2. Bagaimana konsep penerapan ILP dengan metode heuristik untuk mendapatkan biaya proses pengemasan yang minimal?
      3. Bagaimana implementasi konsep tersebut pada C dan Matlab?
      4. Bagaimana melakukan uji metode di atas untuk menyelesaikan permasalahan optimasi pengemasan peti kemas?

# BATASAN MASALAH

Permasalahan yang dibahas dalam Tugas Akhir ini memiliki beberapa batasan, yaitu sebagai berikut:

* + - 1. Sistem perangkat lunak dibangun dengan menggunakan perangkat lunak Dev-C++ 4.9.9.2, Matlab 7.0 dan TOMLAB Optimizaton.
      2. Data uji coba adalah data pengemasan barang yang terdapat dalam paper [[1](#Jounal1)]. Terdapat 50 data uji coba.
      3. Jumlah peti kemas yang tersedia tidak terbatas.

# TUJUAN DAN MANFAAT TUGAS AKHIR

Tugas akhir ini bertujuan untuk :

* + - 1. Mendapatkan pola pengemasan barang yang baik dan efektif.
      2. Mendapatkan biaya proses pengemasan yang minimal.
      3. Mengimplementasikan ILP dengan metode heuristik untuk optimasi pada pengemasan peti kemas.
      4. Membuktikan bahwa ILP dengan menggunakan metode heuristik dapat menyelesaikan permasalahan optimasi biaya pengemasan peti kemas.

# TINJAUAN PUSTAKA

## Integer Linear Programming

Permasalahan *Linear Programming (LP)* adalah optimasi permasalahan yang mencoba memaksimalkan ataupun meminimalkan fungsi linear dari variabel-variabel keputusan [[2](#Book01)]. Fungsi tersebut disebut fungsi objektif. Nilai-nilai dari variabel keputusan yang akan dicari harus memenuhi batasan-batasan yang telah ditentukan. Batasan-batasan tersebut harus berupa persamaan atau pertidaksamaan linear.

*Integer Linear Programming* (*ILP*) adalah pengembangan dari permasalahan *Linear Programming,* yaitu variabel yang digunakan harus berupa bilangan bulat. Dalam pengerjaan Tugas Akhir ini akan digunakan *ILP* untuk menyelesaikan permasalahan optimasi pengemasan peti kemas.

## Pengemasan Peti Kemas

Pengiriman barang dalam peti kemas memiliki peranan sangat penting dalam hampir semua industri. Setiap industri pasti menginginkan pengiriman barang yang efektif dan efisien untuk mendapatkan keuntungan yang maksimal. Agar mendapatkan keuntungan yang maksimal, biaya pengemasan harus seminimal mungkin tetapi memenuhi permintaan yang ada.

Permasalahan yang diangkat dalam Tugas Akhir ini, adalah permasalahan pengemasan barang dalam peti kemas, dimana terdapat beberapa jenis peti kemas yang memiliki ukuran dan biaya pengemasan yang berbeda-beda [[1](#Jounal1)]. Dengan batasan ukuran dan biaya yang ada, diperlukan suatu strategi pengemasan barang dalam peti kemas dengan pola yang baik dan efisien sehingga mampu mengurangi biaya pengemasan yang dibutuhkan.

## Heuristik

Metode Heuristik adalah sebuah pendekatan dalam penyelesaian masalah. Metode ini biasa digunakan ketika suatu algoritma atau metode konvensional cukup lambat atau bahkan tidak dapat mencari solusi yang diinginkan [[3](#Online1)]. Tujuan dari implementasi metode heuristik adalah menghasilkan solusi yang cukup bagus dari suatu permasalahan, walaupun tidak dapat dipastikan solusi tersebut adalah solusi yang terbaik.

Di dalam pustaka [[1](#Jounal1)], disebutkan bahwa implementasi metode heuristik untuk menentukan pola pengemasan barang dalam peti kemas yang baik dan efektif. Dalam pengerjaan Tugas Akhir ini, metode heuristik yang dipakai akan membantu untuk mendekati solusi yang optimal.

## MATLAB

MATLAB adalah lingkungan pengembangan yang ditujukan untuk komputasi teknis, visualisasi, dan pemrograman [[4](#Online2)]. MATLAB adalah sistem interaktif yang mempunyai basis data *array* yang tidak membutuhkan dimensi. Ini memungkinkan kita dapat menyelesaikan banyak masalah komputasi teknis, khususnya yang berkaitan dengan formulasi matrik dan vektor. Pemilihan MATLAB sebagai lingkungan pengembangan dalam pengerjaan Tugas Akhir ini karena MATLAB sangat handal untuk pemodelan data berupa matriks dan vektor, serta keberadaan TOMLAB yang dikembangkan pada MATLAB.

## TOMLAB Optimization

TOMLAB adalah kakas kerja yang dikembangkan pada Matlab untuk pemecahan permasalahan optimasi. Pemodelan pada TOMLAB sangat fleksibel, mudah digunakan, dan sangat handal untuk menyelesaikan berbagai permasalahan optimasi.

TOMLAB menyediakan berbagai macam data tes, serta beberapa *solver* untuk melakukan pemecahan masalah optimasi, seperti: CPLEX, MINLP, MINOS, SOL, dan lain sebagainya [[5](#TOM11)]. Dalam TOMLAB kita hanya perlu melakukan pemodelan satu kali, kemudian dapat menggunakan beberapa *solver* untuk membandingkan performanya.

## Dev C++

Dev C++ kakas kerja yang dikembangkan oleh Bloodshed software untuk bahasa pemrograman C maupun C++ pada sistem operasi Windows [[6](#Online4)]. Kakas kerja ini bersifat *free software* sehingga mudah diperoleh.

# RINGKASAN ISI TUGAS AKHIR

Permasalahan yang diangkat dalam Tugas Akhir ini, adalah permasalahan pengemasan barang dalam peti kemas, dimana terdapat beberapa jenis peti kemas yang memiliki ukuran dan biaya pengemasan yang berbeda-beda [[1](#Jounal1)]. Terdapat dua langkah dalam menyelesaikan permasalahan pengemasan ini. Langkah pertama, menentukan pola pengemasan barang yang baik dan efektif di dalam peti kemas. Langkah kedua, menentukan menggunakan *Integer Linear Programming* (ILP) untuk mendapatkan solusi optimal berupa biaya proses pengemasan yang minimal.

Gambar 8.1 menunjukkan tahapan-tahapan penyelesaian permasalahan optimasi pengemasan peti kemas menggunakan ILP dengan metode heuristik.



Gambar 8.1 Diagram Alir ILP dengan Metode Heuristik untuk Optimasi Pengemasan Peti Kemas

Gambar 8.2 menunjukkan arsitektur perangkat lunak yang akan dibuat.



Gambar 8.2 Arsitektur Perangkat Lunak yang Akan Dibuat

# METODOLOGI

## Penyusunan proposal tugas akhir

Dalam proposal tugas akhir ini, permasalahan yang diangkat adalah permasalahan yang ditemukan dalam pengemasan peti kemas. Dengan adanya batasan ukuran dan biaya pengemasan yang berbeda-beda, setiap industri menginginkan pengemasan peti kemas dengan biaya seminimal mungkin namun tetap memenuhi permintaan yang ada.

## Studi literatur

Untuk menentukan pola pengemasan yang baik, maka terlebih dahulu ditentukan nilai *loading factor* dari peti kemas menggunakan algoritma *Binary Search*.

Kode Sumber 9.1 menunjukkan implementasi algoritma *Binary Search*

|  |
| --- |
| 1. Dapatkan himpunan pola pengemasan P 2. *al ←*0,01, *ah ←*1,0, *al ←ah,* iterasi *←* 0 3. **while** iterasi < maksimum iterasi **do**   B *←*himpunan barang yang akan dimuat ke dalam peti kemas  Selesaikan ILP dan tentukan S adalah daftar yang berisi pola pengemasan yang dipilih yang diurutkan berdasarkan penggunaan ukuran peti kemas   1. **for** semua pola pengemasan yang merupakan elemen dari daftar S **do**   Muat barang ke dalam peti kemas sesuai pola pengemasan  Hapus barang dari himpunan B jika telah dimuat ke dalam peti kemas  **end for**   1. Cm ← peti kemas yang memuat beberapa jenis barang, diurutkan berdasarkan penggunaan ukuran peti kemas 2. Cs ← peti kemas yang memuat satu jenis barang, diurutkan berdasarkan penggunaan ukuran peti kemas 3. Jika B tidak dalam kondisi kosong, lakukan *Insert*(Cm, P, B) 4. Jika B tidak dalam kondisi kosong, lakukan *Insert*(Cs, P, B) 5. Jika B tidak dalam kondisi kosong, maka *al ←a*dan *ah ←a* simpan solusinya 6. *a←* (*ah* + *al*) / 2, iterasi *←* i + 1 7. **end while** |

Kode Sumber 9.1 Kode Sumber Algoritma *Binary Search*

Kode Sumber 9.2Menjelaskan fungsi *Insert* pada algoritma *Binary Search*

|  |
| --- |
| 1. **for** setiap peti kemas bermuatan c C **do** 2. Bc merupakan himpunan barang yang telah dimuat ke dalam peti kemas c 3. Tentukan ILP dari peti kemas C dan union himpunan barang B dan Bc 4. Terapkan pola pengemasan barang dengan p merupakan pola pengemasan yang terbentuk dan Bp adalah himpunan barang pada p 5. Muat ulang peti kemas c sesuai dengan p 6. B *←* B - Bp 7. P *←* P p 8. Jika B kosong, maka **break** 9. **end for** |

Kode Sumber 9.2 Kode Sumber Fungsi *Insert*

Pola pengemasan yang digunakan adalah pola kombinasi *GRASP*.

Langkah 1

Tentukan semua kombinasi yang memungkinkan dari jenis barang dan tempatkan dalam suatu himpunan.

Langkah 2

Untuk setiap kombinasi, berikan nilai kemungkinan yang sesuai dengan ukuran peti kemas dalam semua kombinasi.

Langkah 3

Pilih kombinasi yang memiliki nilai kemungkinan terbesar dan hapuskan kombinasi yang tidak dipilih dari himpunan.

Langkah 4

Dapatkan pola pengemasan dari kombinasi yang dipilih dengan asumsi barang berhasil dimuat ke dalam peti kemas.

Langkah 5

Kurangkan jumlah barang yang berhasil dimuat ke dalam peti kemas dari total barang yang ada.

Langkah 6

Kembali ke langkah 2, hingga jumlah pola pengemasan = 5 atau semua kombinasi telah digunakan.

Formulasi ILP untuk meminimalkan biaya pengemasan

*i* : indeks peti kemas

*j*  : indeks jenis barang

*bij* : jumlah barang

*pi* : pola pengemasan

*ci* : biaya pengemasan peti kemas

*xi* : jumlah pola pengemasan

*a* : *loading factor*

*P* : himpunan pola pengemasan

|  |  |
| --- | --- |
|  | (1) |

Persamaan (1) untuk meminimalkan biaya pengemasan.

**Batasan**

|  |  |
| --- | --- |
|  | (1) |

Batasan (1) memastikan pola pengemasan peti kemas memiliki ruang yang cukup untuk semua jenis barang yang ada.

|  |  |
| --- | --- |
|  | (2) |

Batasan (2) memastikan jumlah pola pengemasan adalah bilangan bulat dan lebih dari sama dengan 0.

## Analisis dan desain perangkat lunak

Dalam tugas akhir ini, terdapat dua langkah. Langkah pertama, menentukan pola pengemasan barang yang baik dan efektif di dalam peti kemas diselesaikan dengan kakas bantu Dev-C++ dengan bahasa pemrograman C. Langkah kedua, menggunakan *Integer Linear Programming* (ILP) untuk mendapatkan solusi optimal berupa biaya proses pengemasan yang minimal diselesaikan dengan Matlab dan TOMLAB Optimizaton.

## Implementasi perangkat lunak

Sistem perangkat lunak dibangun dengan menggunakan kakas bantu Dev-C++ dengan bahasa pemrograman C, Matlab dan TOMLAB Optimizaton.

## Pengujian dan evaluasi

Dalam tugas akhir ini, data uji coba yang digunakan adalah data yang terdapat dalam paper [[1](#Jounal1)]. Terdapat 50 data uji coba.

Data uji coba terdiri dari :

* Data jenis barang, ukuran barang, dan jumlah barang yang akan dikemas ke dalam peti kemas.
* Data ukuran peti kemas dan biaya pengemasannya.

## Penyusunan Buku Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam tugas akhir ini serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat. Sistematika penulisan buku tugas akhir secara garis besar antara lain:

1. Pendahuluan
   1. Latar Belakang
   2. Rumusan Masalah
   3. Batasan Tugas Akhir
   4. Tujuan
   5. Metodologi
   6. Sistematika Penulisan
2. Tinjauan Pustaka
3. Desain dan Implementasi
4. Pengujian dan Evaluasi
5. Kesimpulan dan Saran
6. Daftar Pustaka

# JADWAL KEGIATAN

Pengerjaan tugas akhir akan dilakukan mengikuti rencana pengerjaan seperti yang ditunjukkan oleh Tabel 10.1.

Tabel 10.1 Jadwal Rencana Pengerjaan Tugas Akhir

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Tahapan | 2013 - 2014 | | | | | | | | | | | | | | | | | | | | |
| September | | | | Oktober | | | | November | | | | Desember | | | | | Januari | | | |
| Penyusunan Proposal |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Studi Literatur |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Perancangan sistem |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Implementasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Pengujian dan evaluasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Penyusunan buku |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

# DAFTAR PUSTAKA

|  |  |
| --- | --- |
| [1] | C. H. Che, W. Huang, A. Lim, and Z. Wenbin, "The multiple container loading cost minimization problem," *European Journal of Operational Research*, pp. 501-511, 2011. |
| [2] | F. S. Hiller and G. J. Lieberman, *Introduction to Operations Research*, 7th ed. 2001. |
| [3] | (2011, Sep.) Wikipedia, the free encyclopedia. [Online]. <http://en.wikipedia.org/wiki/Heuristic_%28computer_science%29> |
| [4] | (2011, Sep.) MATLAB- The language of technical computing. [Online]. <http://www.mathworks.com/products/matlab/> |
| [5] | (2011, Sep.) TOMLAB OPTIMIZATION. [Online]. <http://tomopt.com/tomlab/about/> |
| [6] | (2011, Sep.) Bloodshed Software - Dev-C++. [Online]. <http://www.bloodshed.net/devcpp.html> |