

BAB II

TINJAUAN PUSTAKA

2.1 Optimasi

Optimasi adalah suatu pencapaian keadaan atau dengan kata lain tindakan terbaik yang mampu dicapai dari suatu masalah pengambilan keputusan dengan berbagai macam sumber daya yang membatasinya. Optimasi merupakan suatu pencapaian terbaik dari usaha yang telah dilakukan. Optimasi *linear* erat kaitannya dengan bagaimana menentukan nilai-nilai ekstrem pada fungsi *linear* maksimasi atau minimasi. Persoalan optimasi secara umum terbagi menjadi dua yaitu optimasi tanpa kendala dan optimasi dengan kendala. Pada dasarnya optimasi dengan kendala adalah penentuan dari persoalan berbagai nilai variabel suatu fungsi untuk mendapatkan hasil yang maksimum atau minimum dengan memperhatikan batasan-batasan yang ada (Silaban dkk, 2022).

Teknik optimasi sangat aplikatif pada permasalahan-permasalahan yang menyangkut pengoptimalan, baik itu maksimasi atau minimasi. Teknik optimasi merupakan suatu teknik pengalokasian sumber daya, baik bahan baku, waktu, tenaga kerja maupun uang, tergantung dari kondisi yang diinginkan. Dengan menggunakan teknik ini, maka sumber daya terbatas yang dimiliki dapat terproses dengan baik dan mendapatkan hasil yang maksimal. Terdapat banyak jenis teknik optimasi yang dapat digunakan mengikuti proyek yang akan di optimasi yaitu optimasi *linear* atau yang sering disebut program *linear*, kemudian optimasi *linear* dengan variabelnya bilangan *Integer* atau bilangan bulat atau yang sering disebut dengan *Integer Linear Programming*, ada juga optimasi *linear* yang variabelnya bersifat ya atau tidak atau disimbolkan 1 jika ya, 0 jika tidak yang dikenal dengan *Binary Integer Linear Program* (Suwirmayanti, 2017). Terdapat banyak teknik optimasi diantaranya yaitu (Purba dan Ahyaningsih, 2020):

1. Program Linear

Program linear merupakan suatu model dari penelitian operasional (riset operasi) yang digunakan untuk memecahkan masalah optimasi.

2. Metode Simpleks

Metode simpleks dikembangkan oleh George Dantzig pada tahun 1947, berbeda *Linear Programming* dengan metode grafik yang hanya dapat digunakan untuk menyelesaikan kasus dengan dua variabel keputusan, maka metode simpleks dapat digunakan untuk memecahkan kasus dengan banyak variabel. Metode simpleks ialah suatu metode yang secara sistematis dimulai dari suatu pemecahan dasar yang feasible ke pemecahan dasar yang feasible lainnya dan ini dilakukan berulang-ulang

3. *Integer Programming*

Integer programming adalah persoalan program linier (*linear programming*) dimana pemecahan optimalnya harus menghasilkan bilangan bulat (*integer*) jadi bukan pecahan.

2.2 Penjadwalan Kelas

Penjadwalan adalah aspek yang penting dalam pengendalian operasi baik dalam industri manufaktur maupun jasa (Kurnia, 2021). Penjadwalan adalah kumpulan kebijakan dan mekanisme dalam sistem operasi yang berhubungan dengan urutan kerja yang dilakukan sistem komputer. Penjadwalan digunakan untuk memutuskan proses yang harus bekerja serta kapan dan berapa lama proses tersebut berjalan. Penjadwalan dibutuhkan untuk perkuliahan dengan pengalokasian sumber daya yang tepat, seperti ruangan yang digunakan, mahasiswa, dosen, dan mata kuliah. Dengan pengaturan penjadwalan yang efektif dan efisien, universitas akan dapat memenuhi perkuliahan tepat pada waktu serta mata kuliah yang telah ditentukan. Penjadwalan diperlukan ketika beberapa mata kuliah harus dilaksanakan pada suatu ruangan tertentu yang tidak bisa digunakan lebih dari mata kuliah pada saat yang sama. Penjadwalan yang baik akan memaksimalkan efektivitas pemanfaatan sumber daya yang ada, sehingga

penjadwalan merupakan kegiatan yang penting dalam perencanaan dan pengendalian produksi. Tahap perencanaan dan tahap implementasi dari kegiatan penjadwalan merupakan masalah yang kompleks (Lalang dan Aloha, 2022). Sasaran utama proses penjadwalan adalah (Sasongko dkk, 2020):

1. Adil, tidak ada proses yang tidak kebagian layanan.
2. Efisien, pemroses dijaga tetap bekerja agar tidak ada waktu yang terbuang.
3. Waktu tanggap, termasuk di dalamnya sistem waktu interaktif dan sistem waktu nyata.
4. *Turn around time*, waktu yang diperlukan untuk serangkaian proses.
5. *Throughput*, jumlah kerja yang didapat dilakukan dalam satu satuan waktu

Adapun untuk aktivitas penjadwalan pada dasarnya dapat dibedakan menjadi lima tingkatan dengan masing-masing kurun waktunya, yaitu (Ramadhani, 2018):

- a) *Long-range planning* (2 – 5 tahun)
- b) *Middle-range planning* (1 – 2 tahun)
- c) *Short-range planning* (3 – 6 bulan)
- d) Penjadwalan (2 – 6 minggu)
- e) Penjadwalan reaktif atau *kontrol* (1 – 3 hari)

Penjadwalan kelas adalah mengalokasikan sumber daya dari waktu ke waktu untuk melakukan sejumlah tugas atau untuk menghasilkan sebuah sekumpulan pekerjaan. Sumber daya yang dimaksudkan adalah dosen, ruangan kelas, mata kuliah, dan mahasiswa. Masalah mendasar dari penjadwalan ruang kuliah adalah menetapkan kegiatan-kegiatan universitas seperti perkuliahan terhadap berbagai sumber daya tersebut. Kegiatan perkuliahan membutuhkan ruangan untuk menunjang kegiatan setiap aktivitas akademik, seperti ruang kuliah (Kurniawati dan Maskur, 2015).

Penjadwalan perkuliahan (*lecture timetabling*) adalah masalah menempatkan waktu dan ruangan kepada sejumlah kuliah, tutorial, dan kegiatan akademik sejenis, dengan memperhatikan sejumlah aturan yang berhubungan

dengan kapasitas dan lokasi dari ruangan yang tersedia, waktu bebas yang diperlukan dan sejumlah aturan lain yang berkaitan dengan toleransi untuk dosen, dan berhubungan antar mata kuliah khusus (Ruhayat dkk, 2015). Penjadwalan kelas merupakan penyusunan dan pengaturan jadwal mata kuliah - mata kuliah pada slot waktu yang tersedia selama satu minggu beserta pembagian ruang kelasnya seperti ditunjukkan pada penjadwalan tersebut harus memperhatikan beberapa hal seperti kapasitas ruang, dosen, mahasiswa, dan jumlah SKS setiap mata kuliah (Ana dkk, 2013).

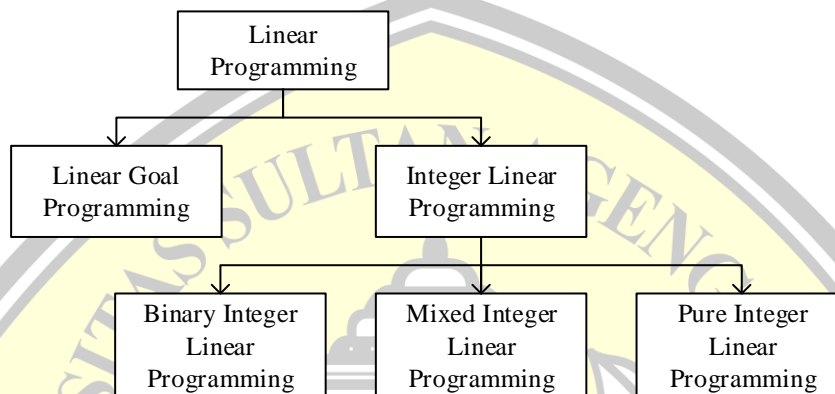
2.3 University Course Scheduling Problem (UCSP)

Permasalahan penjadwalan mata kuliah standar internasional, dikenal dengan istilah *University Course Scheduling Problem (UCSP)*. Penyusunan jadwal mata kuliah - mata kuliah pada slot waktu yang tersedia selama satu minggu. pengaturan jadwal mata kuliah - mata kuliah pada slot waktu yang tersedia selama satu minggu. Komponen yang mempengaruhi jadwal terdiri dari kapasitas ruangan yang tersedia, waktu kehadiran dosen, mahasiswa, serta batasan-batasan yang ditentukan. Batasan atau kendala (*constraint*) yang terdapat dalam UCSP terbagi dalam dua kategori yaitu batasan lunak (*soft constraint*) dan batasan mutlak (*hard constraint*). Batasan mutlak merupakan batasan penting yang harus terpenuhi dalam pembuatan jadwal, sedangkan batasan lunak merupakan batasan yang menentukan kualitas dari jadwal yang dibuat (Ana dkk, 2013).

2.4 Linear Programming (LP)

Program *Linear (Linear Programming)* merupakan teknik aplikasi dari matematika yang dikembangkan oleh (Dantzig, 1990) Kata “linear” berarti bahwa seluruh fungsi persamaan atau pertidaksamaan matematis yang disajikan dari permasalahan ini haruslah bersifat *linear*, sedangkan kata “program” merupakan sinonim untuk model perencanaan. Jadi, program *linear* merupakan perencanaan kegiatan-kegiatan untuk mencapai hasil yang optimal, yaitu suatu hasil yang mencerminkan tercapainya sasaran atau tujuan tertentu yang paling baik. *Linear Programming* digunakan untuk memecahkan masalah-masalah yang memerlukan pemecahan dalam proses maksimasi atau minimasi dengan menggunakan teknik matematik dalam bentuk ketidaksamaan *linear*. Pemecahan masalah dengan

menggunakan *linear programming* akan memperhatikan kendala-kendala tersebut dalam bentuk ketidaksamaan *linear* dalam bentuk variabel-variabel tertentu. *Linear programming* dapat didefinisikan sebagai metode yang digunakan untuk mengombinasikan faktor-faktor produksi yang bertujuan untuk mengoptimalkan suatu tujuan dengan rencana produksi dan peralatan tertentu (Silaban dkk, 2022). Berikut ini adalah perluasan dari *linear programming* (Garfinkel dan Nemhauser, 1972).



Gambar 1. Pengembangan Metode *Linear Programming*

Sumber: Garfinkel dan Nemhauser, 1972

Pada Gambar 1 dapat diketahui bahwa *linear programming* memiliki pengembangan model. Yaitu *goal programming* dan *integer linear programming*. Menurut Harjiyanto (2014) *Goal Programming* adalah salah satu model matematis yang dipandang sesuai digunakan untuk pemecahan masalah multi tujuan karena melalui variabel deviasinya, *goal programming* secara otomatis menangkap informasi tentang pencapaian relatif dari tujuan yang ada. *Integer linear programming* merupakan pengembangan dari program *linear* dimana beberapa atau semua variabel keputusannya harus berupa *integer*. Adapun jenis-jenis *integer linear programming* terdiri dari program integer campuran (*mixed integer programming*), program integer murni (*pure integer programming*) dan *binary integer linear programming* (Garfinkel dan Nemhauser, 1972).

Pemrograman linear merupakan proses penyusunan program linear yang solusinya menjadi dasar bagi pengambilan keputusan terhadap masalah riil yang dimodelkan. Menurut (Mattengnga, 2021) program linear berkaitan dengan penjelasan suatu dunia nyata sebagai suatu model matematika yang terdiri atas

sebuah fungsi tujuan. Pada pemecahan masalah terkait program *linear* terdapat syarat dalam penyusunan model diantaranya yaitu terdapat variabel keputusan (*decision variable*), fungsi tujuan (*objective function*) dan batasan atau kendala (*constraint*) (Jainuddin, 2019). Agar program *linear* dapat diterapkan, beberapa asumsi perlu diterapkan seperti: fungsi tujuan dan persamaan setiap batasan harus linear, parameter-parameter harus diketahui atau dapat diperkirakan dengan pasti (deterministik), dan variabel-variabel keputusan harus dapat dibagi, ini berarti bahwa suatu penyelesaian “*feasible*” dapat berupa bilangan pecahan (Ngusman, 2018).

Menurut (Hilman, 2017) Dalam menyusun model dari formulasi persoalan program linear, digunakan karakteristik-karakteristik yang biasa digunakan dalam program linear yaitu:

1. Variabel Keputusan

Variabel keputusan adalah variabel yang menguraikan secara lengkap keputusan-keputusan yang akan dibuat.

2. Fungsi Tujuan

Fungsi tujuan merupakan fungsi dari variabel keputusan yang akan dimaksimalkan atau diminimalkan sesuai dengan yang sudah ditentukan sebelumnya. Fungsi tujuan ini merupakan formula matematis tentang hal-hal yang ingin dicapai, sehingga untuk mencapai tujuan tersebut harus menghadapi berbagai kendala atau batasan. Menurut Ristianasari (2017) dalam penulisan fungsi tujuan dapat dituliskan dengan formula sebagai berikut:

$$\text{Maximize} = nX1 + nX2 + \dots + nXn \quad (1)$$

$$\text{Minimize} = nX1 + nX2 + \dots + nXn \quad (2)$$

Perumusan *maximize* atau maksimasi digunakan untuk tujuan yang fungsinya adalah meningkatkan tujuan, sedangkan perumusan *minimize* atau minimasi digunakan untuk tujuan yang fungsinya adalah meminimalkan tujuan.

Keterangan:

n = Nilai positif dari variabel.

X_1, X_2, \dots, X_n = Variabel keputusan yang digunakan untuk mencapai fungsi tujuan.

3. Batasan

Batasan merupakan variabel yang menjadi pembatas atau kendala yang dihadapi dalam mencapai fungsi tujuan. Dalam penulisan batasan tersebut perlu memperhatikan mengenai variabel yang menjadi batasan dalam mencapai fungsi tujuan. Menurut Ristianasari (2017) dalam penulisan fungsi tujuan dapat dituliskan dengan formula sebagai berikut:

$$\text{Batasan 1} = nX_1 + nX_2 + \dots + nX_n \geq p \quad (3)$$

$$\text{Batasan 2} = nX_1 + nX_2 + \dots + nX_n \leq q \quad (4)$$

Keterangan:

Batasan 1&2 = Variabel-variabel yang menjadi batasan dalam mencapai fungsi tujuan.

n = Nilai positif dari variabel.

X_1, X_2, \dots, X_n = Variabel keputusan yang digunakan untuk mencapai fungsi tujuan.

p & q = Nilai konstanta yang menjadi pembatas pada masing-masing batasan.

2.4.1 *Integer Linear Programming* (ILP)

Menurut (Anderson dkk, 2004) *Integer linear programming* merupakan model program linear dengan persyaratan tambahan yaitu beberapa atau semua variabel keputusan harus merupakan bilangan bulat. Penggunaan variabel bilangan bulat memberikan tambahan fleksibilitas dalam pembuatan model. Menurut Borcinova (2017) *Integer linear programming* merupakan sebuah metode matematik untuk memaksimalkan profit dan meminimalisasi *cost* berdasarkan sebuah model matematika yang melibatkan variabel-variabel yang bertipe *integer* yang direpresentasikan dalam suatu bentuk hubungan yang bersifat linear. Penyelesaian persamaan matematik biasanya menghasilkan keluaran berupa bilangan pecahan, jika sistem mengharapkan keluaran bilangan *integer*, maka ILP merupakan algoritma paling tepat karena jika menggunakan pembulatan sering kali hasil yang diperoleh bukan solusi optimal. *Integer Programming* (Program

Bilangan Bulat) muncul karena tidak semua variabel keputusan dapat berupa bilangan pecahan, melainkan bilangan bulat. *Integer linear programming* (IP) merupakan suatu pemrograman linear yang sebagian atau semua variabel yang digunakan merupakan bilangan bulat (*integer*). *Pure Integer linear programming* merupakan pengembangan dari program linear dimana semua variabelnya harus berupa integer (Garfinkel dan Nemhauser, 1972). Ada 3 jenis *integer programming* yaitu (Winston, 2004):

- a) *Pure Integer Linear Programming* (PILP), jika suatu *integer programming* menggunakan semua variabel yang berupa *integer*.
- b) *Mixed Integer Linear Programming* (MILP), jika suatu *integer programming* menggunakan sebagian saja variabel yang integer.
- c) *Binary Integer Linear Programming* (BILP) 0-1 IP, jika suatu *integer programming* menggunakan variabel yang bernilai 0 atau 1.

Secara matematis, program bilangan bulat dapat dinyatakan sebagai berikut (Basriati, 2018):

Maksimumkan:

$$z = \sum_{j=1}^n C_j X_j \quad (5)$$

dengan kendala:

$$\sum_{j=1}^n a_{ij} x_j \leq b_i \quad (6)$$

$x_j \geq 0$, *Integer* untuk setiap x_j

Untuk $i = 1, 2, \dots, m$.

2.5 **Branch and Bound**

Branch and bound merupakan salah satu metode untuk menyelesaikan permasalahan terkait program linear dari berbagai permasalahan pada optimasi yang menghasilkan variabel-variabel keputusan bilangan bulat (Andriyani, 2022). Metode ini membatasi penyelesaian optimal yang akan menghasilkan bilangan pecahan dengan cara membuat cabang atas atau bawah bagi masing-masing

variabel keputusan yang bernilai pecahan agar bernilai bulat sehingga setiap pembatasan akan menghasilkan cabang baru yang akan dievaluasi secara sistematis sampai solusi terbaik ditemukan (Suryadi Nasution dkk., 2023) Metode ini sering digunakan untuk menyelesaikan suatu masalah program *integer* karena hasil yang diperoleh dalam penyelesaian optimal lebih teliti dan lebih baik dibanding metode lainnya. Kelemahan pokok metode ini adalah prosedur untuk mencapai hasil optimal yang sangat panjang. Prinsip dasar metode ini yaitu memecah daerah fisibel layak suatu masalah program linear dengan membuat sub-masalah (Andriyani, 2022).

Langkah-langkah dalam menyelesaikan persoalan dengan menggunakan metode *Branch and Bound* adalah sebagai berikut (Basriati, 2018):

1. Menyelesaikan persoalan program linear dengan metode simpleks tanpa batasan *integer*.
2. Memeriksa solusi optimalnya. Jika variabel basis yang bernilai *integer*, maka solusi optimal telah tercapai. Jika tidak bernilai *integer*, maka lanjutkan langkah 3.
3. Memilih variabel yang mempunyai nilai pecahan terbesar (artinya bilangan desimal terbesar) dari setiap variabel untuk dijadikan percabangan ke dalam sub-masalah. Ciptakan dua batasan baru untuk variabel ini, dengan batasan \leq dan batasan \geq .
4. Menjadikan solusi pada penyelesaian langkah 1 sebagai batas atas dan untuk batas bawahnya merupakan solusi yang variabel keputusannya telah dibulatkan.
5. Menyelesaikan model program linear dengan batasan baru yang ditambahkan pada setiap sub-masalah.
6. Suatu solusi *integer* fisibel (layak) adalah sama baik atau lebih baik dari batas atas untuk setiap sub-masalah yang dicari. Jika solusi yang demikian terjadi, suatu sub-masalah dengan batas atas terbaik dipilih untuk dicabangkan.

2.6 Pengembangan Model Matematika

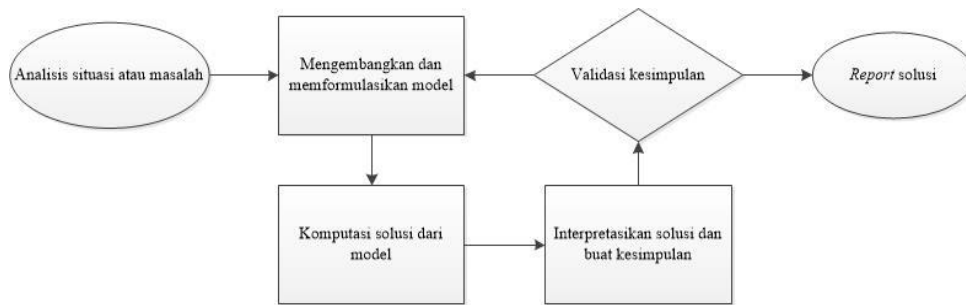
Menurut (Ilhami, 2020) pemodelan merupakan salah satu cara untuk menentukan solusi dari suatu permasalahan di dunia nyata. Pemodelan berfungsi untuk memodelkan fenomena di dunia nyata menjadi model, pada umumnya menggunakan notasi dan fungsi matematika untuk memformulasikan modelnya. Solusi model dapat dilakukan dengan menggunakan berbagai pendekatan seperti pendekatan analitik, *heuristic*, *metaheuristic* dan simulasi. Secara sederhana tahapan pengembangan model matematika diantaranya:

1. Menyusun pertanyaan
2. Menentukan pendekatan model yang tepat
3. Memformulasikan model
4. Mencari solusi dari model
5. Menjawab pertanyaan

Menurut (Daellenbach, 1994), model matematika dapat diartikan sebagai model yang baik jika memenuhi unsur-unsur sebagai berikut:

1. *Simple*, yaitu mudah dipahami oleh pengamat.
2. *Complete* atau komprehensif, yaitu melibatkan semua aspek yang signifikan dalam menentukan fungsi objektif.
3. *Easy to manipulate*, yaitu memungkinkan untuk memperoleh hasil dari model.
4. *Adaptive*, yaitu dapat menyesuaikan dengan perubahan yang terjadi pada masalah.
5. *Easy to communicate*, yaitu mudah untuk diterapkan, diperbaharui dan diubah sesuai situasi.
6. *Appropriate*, yaitu menghasilkan *output* yang sesuai dan *output* tersebut berada dalam horizon waktu yang masuk akal.

Menurut Anhalt dan Cortez (2015) memberikan gambaran terkait siklus pemodelan matematika. Dikatakan siklus karena dalam hal ini siklus model bersifat berulang hingga memperoleh model yang dapat memberikan solusi dalam menyelesaikan permasalahan yang ditentukan.



Gambar 2. Elemen dalam Siklus Pemodelan

Sumber: Anhalt dan Cortez, 2015

Dari Gambar 2 dapat diketahui bahwa setiap aktivitas saling terhubung ke setiap elemen dan membentuk sebuah siklus. Berikut ini adalah deskripsi eksplisit dari siklus pemodelan dapat dilihat pada Tabel 2:

Tabel 2. Elemen Pemodelan Matematika dan Aktivitas yang Dilakukan

No.	Elemen Pemodelan	Aktivitas Detail
1	Analisis situasi atau masalah	Mengidentifikasi permasalahan yang harus diselesaikan atau perlu dipahami dan dijelaskan, dan melakukan penelitian pendahuluan jika diperlukan
2	Mengembangkan dan memformulasikan model	Mengumpulkan informasi yang tersedia dan asumsi (jika diperlukan) untuk kemudian diterjemahkan menjadi permasalahan matematika yang dapat diselesaikan.
3	Komputasi solusi model	Cari solusi dari model matematika, analisis dan lakukan percobaan operasi yang berbeda pada model. Kemudian periksa keakuratan model
4	Interpretasikan solusi dan buat kesimpulan	Interpretasikan solusi matematika ke dalam situasi sesungguhnya, kemudian tarik kesimpulan kesesuaian solusi dengan keadaan sebenarnya
5	Validasi kesimpulan	Refleksikan apakah solusi matematika tersebut masuk akal terhadap keadaan sebenarnya, jika tidak maka kembali pada langkah 2
6	Kembangkan dan formulasikan model baru atau model yang dimodifikasi	Revisi asumsi berdasarkan fakta solusi model, terjemahkan menjadi model baru atau modifikasi model. Jika terjadi modifikasi yang mengubah model maka perlu dilakukan kembali langkah 3 – 5.
7	Laporkan solusi	Berikan jabaran kesimpulan serta penjelasan alasan kesimpulan tersebut

Sumber: Anhalt dan Cortez, 2015

2.7 Lingo

Lingo adalah sebuah *software* yang dapat digunakan untuk mencari penyelesaian dari permasalahan dalam pemrograman *linear*, *non-linear* dan *integer*. *Lingo* merupakan generasi yang lebih tua dari *software lindo*. Dengan menggunakan *software Lingo* pengguna memungkinkan melakukan perhitungan permasalahan *linear programming* dengan jumlah n variabel. Cara kerja *Lingo* tidak jauh berbeda dengan *lindo* yaitu memasukkan data dengan model *linear* yang dibuat, kemudian menaksirkan kebenaran dan kelayakan data berdasarkan penyelesaiannya (Kurnia dkk, 2020). Cara menggunakan *software Lingo* terdiri dari beberapa tahapan yang perlu dilakukan diantaranya (Harjiyanto, 2014):

1. Merumuskan masalah dalam kerangka program linear;
2. Menuliskan dalam persamaan matematik;
3. Merumuskan rumusan ke dalam *LINGO* kemudian mengeksekusi rumus tersebut;
4. Interpretasi *output LINGO*.

Software Lingo menyediakan paket integrasi lengkap yang termasuk di dalamnya yaitu bahasa untuk optimasi model yang mudah dipahami. Selain itu *Lingo* juga dapat digunakan dalam pengambilan keputusan seperti perencanaan produksi, transportasi, keuangan, alokasi saham, penjadwalan, inventarisasi, pengaturan model, alokasi daya dan lain sebagainya. Sebuah optimasi terdiri dari tiga bagian utama yaitu (Mustika dan Syafi'i Ceffi, 2020):

1. Fungsi tujuan yaitu sebuah formula yang mendeskripsikan apa yang harus di optimasi dalam suatu model. Sebagai contoh, fungsi tujuan dari suatu model adalah maksimasi keuntungan.

Contoh penulisan model matematik pada *software Lingo*:

```
min = @sum(links(i,j): cost(i,j)*volume(i,j));
```

```
@for(d(j): @sum(s(i):volume(i,j)) >= demand(j));
```

```
@for(s(i): @sum(d(j):volume(i,j)) <= supply(i)); End
```

2. Variabel adalah kuantitas yang bisa diubah untuk mengeluarkan hasil yang optimal dari fungsi tujuan. Contoh penulisan model matematis untuk variabel pada *software Lingo*:

s / s1 s2 s3 s4/: supply;

d / d1 d2 d3 d4 d5 d6/: demand;

links(s, d):cost, volume;

endsets

3. Batasan formula yaitu yang didefinisikan sebagai nilai pembatas dari suatu variabel. Berikut merupakan contoh model matematis penulisan

batasan formula pada *software Lingo*:

supply = 32000 10000 3500 23000;

demand = 1500 1700 30000 6000 8000 10000;

cost = 11 8 2 3 12 10 11 5 3 2 10 5 14 8 12 10 2 11 11 9 10 5 11 2;

enddata

