



MODUL PERKULIAHAN

Operation Research

Linear Programming, Queue Theory,
Simulation, Decision Tree &
Decision Analysis

Fakultas
Ilmu Komputer

Program Studi
Sistem Informasi

Tatap Muka

Kode MK
18032

Disusun Oleh
Indrajani, S.Kom., MM.

06

Abstract

Membahas Linear programming, queue theory, simulation, decision tree, decision analysis

Kompetensi

Mahasiswa mampu memahami linear programming, queue theory
Mahasiswa mampu menjelaskan simulation, implementasi decision tree dan decision analysis
Mahasiswa mampu menjawab dan menyelesaikan tugas dengan tepat.

1. Linear Programming

Linear programming adalah suatu metode/ teknis matematis yang digunakan untuk membantu manajer dalam pengambilan keputusan.

Tujuan Linear Programming adalah mencari pemecahan persoalan-persoalan yang timbul dalam perusahaan, yaitu mencari keadaan yang optimal dengan memperhitungkan batasan-batasan yang ada.

Linear programming didukung oleh berbagai macam asumsi yang menjadi tulang punggung model tersebut. Asumsi tersebut adalah sebagai berikut :

1. Proportionality

Asumsi ini berarti bahwa naik turunnya nilai z dan penggunaan faktor-faktor produksi yang tersedia akan berubah secara sebanding (proposional) dengan perubahan tingkat kegiatan.

2. Additivity

Asumsi ini berarti bahwa nilai tujuan tiap kegiatan tidak saling mempengaruhi, atau dalam linear programming dianggap bahwa kenaikan nilai tujuan yang diakibatkan oleh kenaikan suatu kegiatan dapat ditumbuhkan tanpa mempengaruhi nilai Z yang diperoleh dari kegiatan lain.

3. Divisibility

Asumsi ini mengatakan bahwa keluaran (output) yang dihasilkan oleh suatu kegiatan dapat berupa bilangan pecahan, demikian pula nilai Z yang dihasilkan.

4. Deterministic (certainty)

Asumsi ini mengatakan bahwa semua parameter yang terdapat dalam model linear programming (a_{ij} , b_j , c_j) dapat diperkirakan dengan pasti meskipun jarang digunakan tepat.

Dalam model linear programming dikenal 2 macam fungsi :

1. Fungsi Tujuan (objective Function)

Fungsi tujuan merupakan fungsi yang menggambarkan tujuan atau sasaran didalam permasalahan linear programming yang berkaitan dengan peraturan secara optimal sumber daya – sumber daya untuk memperoleh keuntungan maksimal.

2. Fungsi Batasan (Constraint Function)

Fungsi merupakan bentuk penyajian secara sistematis batasan-batasan kapasitas yang tersedia akan dialokasikan secara optimal.

Masalah linear programming dapat dinyatakan sebagai proses optimisasi suatu fungsi tujuan dalam bentuk : Memaksimumkan atau meminimumkan

$$Z = C_1 X_1 + C_2 X_2 + \dots + C_n X_n$$

Dengan mengingat batasan-batasan sumber daya

dalam bentuk: $A_{11}X_1 + A_{12}X_2 + \dots +$

$$A_{1n}X_n < B_1$$

$$A_{21}X_1 + A_{22}X_2 + \dots + A_{2n}X_n < B_2$$

$$A_{m1}X_1 + A_{m2}X_2 + \dots + A_{mn}X_n < B_m$$

$$\text{Dan } X_1 > 0, X_2 > 0, \dots, X_n > 0$$

Dimana C_j , A_{ij} dan B_i adalah masukan konstan yang sering disebut sebagai parameter model. Keterangan

M = macam-macam batasan sumber atau fasilitas yang tersedia

N = macam aktivitas yang menggunakan atau fasilitas tersebut
 i = nomor setiap macam sumber atau fasilitas yang tersedia

j = nomor setiap macam aktivitas yang menggunakan sumber atau fasilitas yang tersedia

X_j = tingkat aktivitas kegiatan atau variable keputusan.

A_{ij} = banyaknya sumber daya i yang diperlukan untuk menghasilkan setiap unit output kejadian j

B_i = banyaknya sumber atau fasilitas i yang tersedia untuk dialokasikan ke setiap jenis aktivitas.

Z = nilai fungsi tujuan atau nilai yang dimaksimumkan atau diminimumkan.

C_j = sumbangan per unit kegiatan j

Pada masalah maksimisasi C_j menunjukkan keuntungan atau penerimaan per unit, pada kasus minimisasi C_j menunjukkan biaya per unit.

Beberapa hal yang perlu diperhatikan dalam penerapan linear programming adalah:

- a. Fungsi tujuan dan persamaan setiap batasan harus linear. Ini mencakup pengertian bahwa perubahan nilai z dan penggunaan sumber daya terjadi secara proporsional dengan tingkat perubahan kegiatan
- b. Parameter-parameter harus dapat diketahui/ diperkirakan dengan pasti (deterministic)
- c. Variabel-variabel keputusan harus dapat dibagi ini berarti bahwa suatu penyelesaian "feasible" dapat berupa bilangan pecahan.

Dasar-dasar umum Linear Programming meliputi bentuk model dan prosedur penyelesaian yang dibagi atas dua pemecahan masalah, yaitu:

A. Metode Grafik (Graphical Method)

Metode grafik adalah metode yang digunakan untuk memecahkan masalah linear programming yang menyangkut dua variabel keputusan.

Didalam penerapan metode grafik, ada langkah-langkah yang harus ditempuh adalah sebagai berikut:

- ✓ Menyusun permasalahan yang ada
- ✓ Menentukan fungsi tujuan yang akan dicapai
- ✓ Mengidentifikasi kendala-kendala, yang berlaku dalam bentuk ketidaksamaan menjadi bentuk persamaan
- ✓ Menggambar masing-masing garis pembatas dalam satu system koordinat.
- ✓ Menentukan daerah (area) yang memenuhi batasan-batasan tersebut. Daerah ini disebut dengan "daerah Feasible".

B. Metode Simpleks (Simplex Method)

Metode Simpleks digunakan untuk memecahkan masalah linear programming yang menyangkut dua atau lebih variabel keputusan. Metode simpleks merupakan algoritma untuk memecahkan masalah umum linear programming.

Metode simpleks adalah metode aljabar yang digunakan untuk memecahkan masalah-masalah linear yang terdiri dari tiga variabel atau lebih. Pedoman langkah-langkah penyelesaian metode simplek dapat disusun sebagai berikut:

- 1) Menentukan fungsi objective atau fungsi tujuan yang akan dicapai.
- 2) Mengidentifikasi kendala-kendala (constrain) dalam bentuk ketidaksamaan.
- 3) Merubah ketidaksamaan dari kendala-kendala yang ada menjadi bentuk persamaan, dengan cara menambahkan unsur-unsur slack variable kedalamnya. Suatu variable "slack" menyajikan secara perhitungan jumlah yang diperlukan untuk merubah tanda ketidaksamaan ($<$) menjadi persamaan ($=$) sehingga semua variable ditunjukkan dalam persamaan, setiap variable slack yang tidak berhubungan dengan salah satu persamaan batasan diberi koefisien nol dan ditambahkan ke persamaan itu.
- 4) Memasukan atau menyusun fungsi tujuan dan kendala yang ada ke dalam table simpleks pertama.
- 5) Mencari nilai Z_j , nilai Z_j menunjukkan jumlah laba kotor yang dihasilkan melalui pemasukan satu unit variable ke dalam penyelesaian.
- 6) Menemukan kolom kunci, baris kunci, dan nomor kunci. Kolom kunci ditentukan dengan cara memilih nilai baris $C_j - Z_j$ yang positif terbesar. Dipilih positif terbesar karena permasalahanya adalah maksimisasi. Untuk menentukan baris kunci, terlebih dahulu harus dicari angka-angka pengganti. Angka-angka pengganti merupakan angka-angka hasil bagi antara angka pada kolom kuantitas dengan angka pada kolom kunci yang bersesuaian. Selanjutnya baris kunci dipilih, yaitu baris yang mempunyai angka pengganti yang merupakan positif terkecil.
- 7) Mengganti angka-angka pada baris kunci dengan angka-angka baru. Angka-angka baru diperoleh dengan cara membagi semua angka yang ada pada baris kunci dengan nomor kunci.

- 8) Mengganti angka-angka baru pada baris lain dengan rumus sebagai berikut :

$$\begin{array}{c} \text{Angka} \\ \text{pada} \\ \text{baris lama} \end{array} \quad -- \quad \begin{array}{c} \text{angka pada baris} \\ \text{lama yang ada diatas atau} \\ \text{dibawah nomor kunci} \end{array} \quad \times \quad \begin{array}{c} \text{angka pada baris} \\ \text{kunci yang baru} \\ \text{yang bersesuaian} \end{array} \quad = \quad \begin{array}{c} \text{angka} \\ \text{pada baris} \\ \text{baru} \end{array}$$

- 9) Memasukan atau menyusun angka-angka baru tersebut ke dalam tabel simpleks yang kedua, kemudian mencari nilai Z_j yang baru dan mencari nilai $C_j - Z_j$ masih ada angka positif (lebih besar sama dengan nol), maka dilakukan lagi langkah-langkah dari langkah 6. Jika angka-angka pada baris $C_j - Z_j$ sudah tidak ada lagi yang positif (lebih kecil sama dengan nol) berarti bahwa kombinasi yang dicari sudah optimum.

2. Queue Theory

Teori antrian mengacu kepada pengamatan matematis dan fisik dari suatu kelompok masalah yang ditandai dengan ciri-ciri :

- ✓ Ada masukan dari satuan yang memasuki sistem
- ✓ Satuan yang bergerak melewati sistem adalah diskrit.
- ✓ Satuan yang mulai membutuhkan pelayanan disusun dengan satu cara dan menerima pelayanan menurut susunan tadi.
- ✓ Mekanisme yang ada yakni yang mengatur kapan satu satuan yang melayani selesai dilayani.
- ✓ Paling tidak satu dari dua mekanisme, kedatangan atau pelayanan, tidak ditentukan seluruhnya tetapi dapat diperhitungkan pada satu jenis sistem probabilistik (berpeluang).

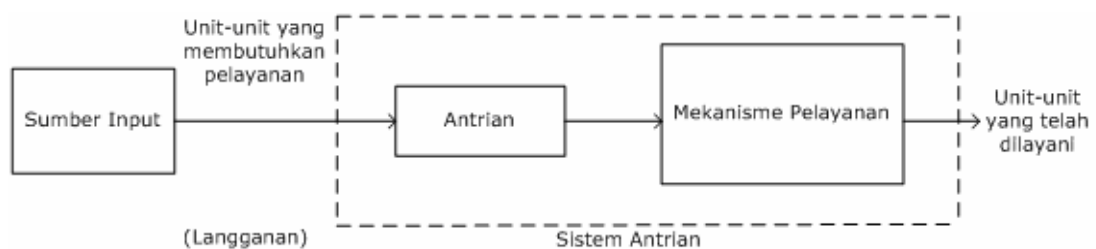
Apabila pelayanan terlalu banyak maka akan memerlukan ongkos yang besar, sebaliknya jika kapasitas pelayanan kurang maka akan terjadi baris penungguan dalam waktu yang cukup lama yang juga akan menimbulkan ongkos baik berupa ongkos sosial, kehilangan langganan ataupun pengangguran kerja. Yang menjadi tujuan utama teori antrian ialah mencapai keseimbangan antara ongkos

pelayanan dengan ongkos yang disebabkan oleh adanya waktu menunggu tersebut.

Ada dua kondisi yang dijumpai dalam sistem manusia-mesin :

- Siklus waktu kegiatan permesinan dan kegiatan pelayanan berlangsung secara konstan dan dapat diprediksikan
- Siklus waktu kegiatan permesinan dan kegiatan pelayanan berlangsung secara random

Proses pada model antrian adalah sebagai berikut:



3. Pohon Keputusan (decision tree)

Pohon keputusan adalah alat pendukung keputusan yang menggunakan grafik seperti pohon atau model keputusan dan konsekuensi yang mungkin terjadi, peluang dari setiap kejadian, biaya sumber daya, dan utilitas.

Pohon Keputusan adalah diagram alir seperti struktur di mana simpul internal merupakan tes pada atribut, setiap cabang merupakan hasil pengujian dan setiap simpul daun merepresentasikan label kelas (keputusan diambil setelah menghitung semua atribut). Setiap jalur dari akar ke daun mencerminkan klasifikasi.

Di dalam analisis keputusan, pohon keputusan dan pengaruh diagram terkait erat digunakan sebagai alat pendukung keputusan visual dan analitis, di mana nilai-nilai yang diharapkan (atau utilitas yang diharapkan) yang menjadi alternatif dihitung.

Sebuah pohon keputusan terdiri dari 3 jenis node :

Keputusan node - biasanya diwakili oleh kotak

Node kesempatan - diwakili oleh lingkaran

Akhir node - diwakili oleh segitiga

Pohon keputusan yang umum digunakan dalam riset operasi, khususnya dalam analisis keputusan, untuk membantu mengidentifikasi strategi yang paling mungkin untuk mencapai tujuan. Jika keputusan harus diambil secara online tanpa menghiraukan

bahwa pengetahuan tidak lengkap, pohon keputusan harus disejajarkan dengan model probabilitas sebagai model pilihan terbaik atau algoritma pemilihan model online. Penggunaan lain pohon keputusan adalah sebagai sarana deskriptif untuk menghitung probabilitas bersyarat.

Kelebihan dari metode pohon keputusan adalah:

- Daerah pengambilan keputusan yang sebelumnya kompleks dan sangat global, dapat diubah menjadi lebih simpel dan spesifik.
- Eliminasi perhitungan-perhitungan yang tidak diperlukan, karena ketika menggunakan metode pohon keputusan maka sample diuji hanya berdasarkan kriteria atau kelas tertentu.
- Fleksibel untuk memilih fitur dari internal node yang berbeda, fitur yang terpilih akan membedakan suatu kriteria dibandingkan kriteria yang lain dalam node yang sama. Kefleksibelan metode pohon keputusan ini meningkatkan kualitas keputusan yang dihasilkan jika dibandingkan ketika menggunakan metode penghitungan satu tahap yang lebih konvensional
- Dalam analisis multivariat, dengan kriteria dan kelas yang jumlahnya sangat banyak, seorang penguji biasanya perlu untuk mengestimasi baik itu distribusi dimensi tinggi ataupun parameter tertentu dari distribusi kelas tersebut. Metode pohon keputusan dapat menghindari munculnya permasalahan ini dengan menggunakan kriteria yang jumlahnya lebih sedikit pada setiap node internal tanpa banyak mengurangi kualitas keputusan yang dihasilkan.

Kelemahan Pohon keputusan

- Terjadi overlap terutama ketika kelas-kelas dan kriteria yang digunakan jumlahnya sangat banyak. Hal tersebut juga dapat menyebabkan meningkatnya waktu pengambilan keputusan dan jumlah memori yang diperlukan.
- Pengakumulasian jumlah eror dari setiap tingkat dalam sebuah pohon keputusan yang besar.
- Kesulitan dalam mendesain pohon keputusan yang optimal.
- Hasil kualitas keputusan yang didapatkan dari metode pohon keputusan sangat tergantung pada bagaimana pohon tersebut didesain.

4. Simulasi

Simulasi adalah suatu proses peniruan dari sesuatu yang nyata beserta keadaan sekelilingnya (state of affairs). Aksi melakukan simulasi ini secara umum menggambarkan sifat-sifat karakteristik kunci dari kelakuan sistem fisik atau sistem yang abstrak tertentu.

Simulasi adalah sebuah teknik untuk melakukan eksperimen dengan sebuah komputer pada sebuah model dari sebuah sistem manajemen. Simulasi merupakan model DSS yang paling umum digunakan. Simulasi merupakan suatu model deskriptif. Tidak ada pencarian otomatis untuk suatu solusi yang optimal. Model simulasi menggambarkan atau memprediksi karakteristik suatu sistem di bawah kondisi yang berbeda. Proses simulasi biasanya mengulangi sebuah eksperimen, berkali-kali untuk mendapatkan estimasi mengenai efek keseluruhan dari tindakan-tindakan.

Karakteristik Utama Simulasi

Simulasi bukanlah sejenis model biasa; model umumnya merepresentasikan kenyataan, sedangkan simulasi biasanya menirukan kenyataan tersebut. Singkatnya, ini berarti ada sedikit penyederhanaan kenyataan dalam model simulasi dibandingkan dengan jenis model lainnya.

Simulasi adalah teknik untuk melaksanakan percobaan. Artinya, simulasi melibatkan testing pada nilai-nilai tertentu dari decision atau uncontrollable variables yang ada pada model dan mengamati akibatnya pada variabel output.

Simulasi lebih bersifat deskriptif (menjelaskan) daripada tool normatif; sehingga tak ada pencarian otomatis untuk solusi optimal. Lebih dari itu, simulasi menjelaskan dan/atau memperkirakan karakteristik sistem tertentu pada pelbagai keadaan yang berbeda-beda. Sekali karakteristik ini diketahui, alternatif terbaik dari alternatif yang ada dapat dipilih.

Simulasi digunakan bilamana permasalahan yang ada terlalu kompleks/sulit bila diselesaikan dengan teknik optimasi numerik (misalnya LP). Kompleksitas disini berarti bahwa permasalahan tadi tak bisa dirumuskan untuk optimasinya atau perumusannya terlalu kompleks.

Keuntungan Simulasi

1. Teori simulasi relatif mudah dan bisa langsung diterapkan
2. Model simulasi mudah untuk menggabungkan pelbagai hubungan dasar dan ketergantungannya.
3. Simulasi lebih bersifat deskriptif daripada normatif. Ini memungkinkan manajer untuk menanyakan jenis pertanyaan "what-if". Sehingga, manajer yang

memiliki pendekatan trial-and-error dalam menyelesaikan masalah dapat melakukannya lebih cepat dan murah, dengan resiko yang lebih kecil, menggunakan bantuan simulasi dan komputer (sebagai pembanding adalah pendekatan trial-and-error dalam sistem nyata).

4. Model simulasi yang akurat membutuhkan knowledge yang dalam dari suatu masalah, yang memaksa MSS builder untuk selalu berkomunikasi dengan manajer.
5. Modelnya dibangun berdasarkan perspektif manajer dan berada dalam struktur keputusannya.
6. Model simulasi dibangun untuk satu permasalahan tertentu, dan biasanya tak bisa menyelesaikan permasalahan yang lain.
7. Simulasi dapat mengatasi variasi yang berbeda-beda dalam pelbagai jenis masalah seperti halnya inventory dan staffing, demikian juga pada fungsi tingkat tinggi manajerial seperti rencana jangka panjang. Sehingga ungkapan untuknya adalah “selalu ada” jika manajer sedang membutuhkannya.
8. Manajer dapat melakukan eksperimen dengan pelbagai variabel yang berbeda untuk menentukan mana yang penting, dan dengan pelbagai alternatif yang berbeda untuk mencari yang terbaik.
9. Simulasi secara umum mengijinkan kita memasukkan kompleksitas kehidupan nyata dari suatu masalah; penyederhanaan tak diperlukan disini. Sebagai contoh, simulasi dapat memanfaatkan distribusi probabilitas kehidupan nyata daripada mengira-ira distribusi teoritis.
10. Sebagai sifat alamiah simulasi, kita dapat menghemat waktu.
11. Mudah untuk mendapatkan pelbagai pengukuran kinerja yang berbeda-beda secara langsung dari simulasi.

Kerugian Simulasi:

- ✎ Tak menjamin solusi yang optimal.
- ✎ Membangun model simulasi seringkali memakan waktu lama dan membutuhkan biaya.
- ✎ Solusi dan inferensi dari satu kasus simulasi biasanya tak bisa ditransfer ke permasalahan yang lain.

- ☞ Simulasi terkadang begitu mudah diterima oleh manajer sehingga solusi analitis yang dapat menghasilkan solusi optimal malah sering dilupakan.

Metodologi Simulasi.

- Definisi masalah.
- Membangun model simulasi.
- Testing dan validasi model.
- Desain percobaan.
- Melakukan percobaan.
- Evaluasi hasil.Implementasi.

Daftar Pustaka

1. Turban, Efraim & Jay E.Aronson, "Decision Support Systems and Intelligent Systems 7th edition", Prentice Hall, 2005.
2. Marakas, George M., "Decision Support Systems in The 21st Century", Prentice Hall, 2003.
3. Mallach, Efraim G., "Decision Support and Data Warehouse Systems", McGraw-Hill International Editions, 2000.
4. Taylor, Bernard W, "Introduction to Management Science 7th edition", Prentice Hall, 2001.