

Pengantar

Riset

Operasional

Pengantar Riset Operasional

Dr. Cris Kuntadi, SE, MM, CA, CPA, PIA, QIA, CMA, CGMA,
ACPA, CIPSAS, CFrA, QRGP, CHRM, Ak.

Dr. TRI YULAELI, SPd, MAk, Ak.

Dr. (C) Rachmat Pramukty, SE, MSi.



Penerbit IPB Press
Jalan Taman Kencana No. 3,
Kota Bogor -Indonesia

C.01/09.2024

Judul Buku:

Pengantar Riset Operasional

Penulis:

Budi Agustono, Suprayitno, Herestina Dewi, Pdt. Juandaha Raya Purba,
Hisarma Saragih, Kaiman Turnip, Setia Dermawan Purba

Penyunting Bahasa:

Abyan

Desain Sampul & Penata Isi:

Alfyandi

Jumlah Halaman:

164 + 10 halaman romawi

Edisi/Cetakan:

Cetakan 1, September 2024

Dicetak dan Diterbitkan oleh

PT Penerbit IPB Press

Anggota IKAPI

Jalan Taman Kencana No. 3, Bogor 16128

Telp. 0251 -8355 158 E-mail: ipbpress@apps.ipb.ac.id

www.ipbpress.com

eISBN:

© 2024, HAK CIPTA DILINDUNGI OLEH UNDANG-UNDANG

Dilarang mengutip atau memperbanyak sebagian atau seluruh isi buku
tanpa izin tertulis dari penerbit

Kata Pengantar

Puji syukur kami panjatkan atas kehadiran Tuhan Yang Maha Esa atas segala rahmat dan karunia-Nya sehingga kami dapat menyelesaikan karya tulis berupa buku ini dengan baik dan tanpa suatu kendala berarti.

Buku berjudul “ Pengantar Riset Operasional ” ini disusun untuk menambah pengetahuan tentang Riset Operasional. Kami memohon maaf apabila terdapat kesalahan dan kekurangan dalam penyusunan makalah ini. Karenanya, kami menerima kritik serta saran yang membangun dari pembaca agar kami dapat menulis makalah secara lebih baik pada kesempatan berikutnya. Besar harapan kami makalah ini dapat bermanfaat dan berdampak besar sehingga dapat memberi inspirasi bagi para pembaca.

Bekasi, 19 September 2023

Daftar Isi

Kata Pengantar	v
Daftar Isi	vii
Bab 1 Pengantar Operations Research	1
1.1 Pengertian <i>Operations Research</i>	1
1.2 Sejarah <i>Operations Research</i>	2
1.3 Perkembangan <i>Operations Research</i>	4
1.4 Definisi <i>Operations Research</i> Menurut Para Ahli	6
1.5 Tahapan Studi <i>Operations Research</i>	7
1.6 Dampak <i>Operations Research</i>	8
1.7 Model Manfaat <i>Operations Research</i>	9
1.8 Langkah dan Tujuan <i>Operations Research</i>	12
1.9 Penerapan dalam Kehidupan Sehari–hari	12
1.10 Metodologi <i>Operation Research</i>	13
1.11 Soal dan Kasus	15
Bab 2 Linear Programming	21
2.1 Pendahuluan Pengertian, Penjelasan dan Contoh	21
2.2 Bentuk Umum Program Linear	23
2.3 Solusi Pada Program Linear	38
2.4 Masalah Teknis dalam Program Linear Metode Grafik	39
2.5 Soal dan Kasus	42

Bab 3 Program Linear Metode Simpleks	47
3.1 Metode Simpleks	47
3.2 Bentuk Baku Model Linear Programming	53
3.3 Solusi pada Program Linear Metode Simpleks	55
3.4 Beberapa Masalah Teknis dalam Program Linier Metode Simpleks	61
3.5 Soal dan Kasus	67
Bab 4 Sistem Operasi dan Sistem Utility	73
4.1 Pendahuluan	73
4.2 Pengertian Sistem Operasi dan Sistem Utility	73
4.3 Pengertian Perangkat Lunak Utility	75
4.4 Perkembangan Sistem Operasi	76
4.5 Fungsi-fungsi Sistem Operasi	78
4.6 Fungsi-fungsi Utility	80
4.7 Jenis-jenis Perangkat Lunak Sistem Operasi	82
4.8 Jenis-jenis Perangkat Lunak Utility	84
4.9 Struktur pada Program Utility	86
4.10 Cara Kerja Sistem Operasi	88
4.11 Sistem Operasi Masa Depan	90
4.12 Soal dan Kasus	92
Bab 5 Transportasi	99
5.1 Pendahuluan Pengertian, Penjelasan dan Contoh	99
5.2 Definisi Model Transportasi menurut Para Ahli	100
5.3 Manfaat dan Tujuan Transportasi	101
5.4 Penerapan Model Transportasi	102
5.5 Asumsi Model Masalah Transportasi	102
5.6 Formulasi Model Masalah Transportasi	103
5.7 Solusi Layak Basis Awal	104

5.8 Metode VAM (Vogel Approximate Method)	104
5.9 Metode MVAM (Modified Vogel's Approximate Method) . .	107
5.10 Solusi Optimal Solusi Optimal.	108
5.11 Soal dan Kasus	111
5.12 Soal dan Kasus 2.	112
Bab 6 Theory Permainan	115
6.1 Pendahuluan	115
6.2 Definisi Theory Permainan Menurut Beberapa Ahli.	119
6.3 Manfaat Theory Permainan	120
6.4 Strategi Murni (Pure Strategy Game)	120
6.5 Strategi Campuran (Mixed Strategy Game)	122
6.6 Aturan Dominansi pada Theory Permainan	123
6.7 Prosedur Penyelesaian Grafik pada Theory Permainan	124
6.8 Penyelesaian Theory Permainan dengan Program Linear	127
6.9 Soal dan Kasus	129
Bab 7 Teori Dualitas	135
7.1 Pendahuluan	135
7.2 Masalah Primal-Dual Asimetrik	139
Bab 8 Teori Analisis Sensitivitas	143
8.1 Pendahuluan	143
8.2 Linear Programming.	144
Bab 9 Penugasan dan Software POM.	149
9.1 Pendahuluan	149
9.2 Penjelasan.	150
9.3 Masalah Penugasan.	154
Daftar Pustaka.	161

Bab 1

Pengantar Operations Research

1.1 Pengertian Operations Research

Secara umum pengertian *research* (riset) dapat diartikan sebagai suatu proses yang terorganisasi dalam mencari kebenaran akan masalah. Sedangkan kata operations (operasi) didefinisikan sebagai tindakan-tindakan yang diterapkan pada beberapa masalah. Dalam kenyataannya sangat sulit untuk mendefinisikan riset operasi secara tegas karena batas-batasnya tidak jelas. Riset Operasi telah banyak didefinisikan oleh para ahli, namun hanya beberapa yang bias digunakan dan diterima secara umum. Morse dan Kimball (1951), mendefinisikan Riset Operasi sebagai metode ilmiah yang memungkinkan para manajer mengambil keputusan mengenai kegiatan yang mereka tangani dengan dasar kuantitatif. Sedangkan Churchman, Arkoff dan Arnoff (1957) mengemukakan pengertian riset operasi sebagai aplikasi metode-metode, teknik-teknik, dan peralatan-peralatan ilmiah dalam menghadapi masalah-masalah yang timbul didalam operasi perusahaan dengan tujuan ditemukannya pemecahan yang optimum masalah-masalah tersebut. Miller dan M.K.

Star (1960) mengartikan Riset Operasi sebagai peralatan manajemen yang menyatukan ilmu pengetahuan, matematika, dan logika dalam kerangka pemecahan masalah-masalah yang dihadapi sehari-hari, sehingga akhirnya permasalahan tersebut dapat dipecahkan secara optimal. Dari definisi di atas dapat disimpulkan bahwa Riset Operasi berkenaan dengan pengambilan keputusan optimal dalam penyusunan model dari sistem-sistem baik deterministic maupun probabilistic yang berasal dari kehidupan nyata. Aplikasi-aplikasi ini, yang terjadi dalam pemerintah, bisnis, teknik, ekonomi, serta ilmu pengetahuan alam dan social ditandai dengan kebutuhan untuk mengalokasikan sumberdaya-sumberdaya yang terbatas.

1.2 Sejarah *Operations Research*

Para ahli sejarah menyebutkan bahwa praksis riset operasi sudah mulai dilakukan semenjak masa Romawi kuno. Praksis tersebut tampak pada analisis yang dilakukan oleh Archimedes untuk melawan blokade laut yang dilakukan oleh armada Syracuse (212 SM). Praksis riset operasi terjadi lagi sesaat menjelang Perang Dunia I, ketika FW. Lancaster, seorang peneliti Inggris, yang mencoba menganalisis hubungan matematis antara kekuatan masing-masing pihak yang bertikai dengan kemungkinan hasil perang. Praksis lain yang juga sering disebut-sebut memiliki karakteristik sebagai sebuah praksis riset operasi juga dilakukan dalam studi Thomas Alva Edison yang mempelajari perang anti kapal selam. Walaupun kurang memiliki gaung pada zamannya, berbagai praksis tersebut di atas merupakan contoh bahwa pendekatan metode ilmiah dapat dipakai untuk memecahkan berbagai masalah, bahkan masalah peperangan.

Perang Dunia II adalah yang menjadi katalisator terbangunnya riset operasional sebagai sebuah keilmuan yang lebih mapan. Pada tahun 1940, sekelompok peneliti yang dipimpin oleh PMS Blackett dari the University of Manchester melakukan studi tentang “Sistem Radar Baru Anti Pesawat Terbang”. Kelompok peneliti ini sering dijuluki sebagai Kelompok Sirkus Blackett (Blackett’s circus). Julukan ini tampaknya lebih didasarkan pada keragaman anggota kelompok peneliti yang berasal dari berbagai disiplin ilmu. Kelompok peneliti Blackcett terdiri atas 3 orang ahli fisiologi, 2 orang ahli matematika, 1 orang ahli astronomi, 1 orang tentara, 1 orang surveyor, 1 orang ahli fisika, dan 2 orang ahli matematika fisika. Keragaman ini merupakan hal yang sangat luar biasa pada waktu itu. Pada tahun 1941, kelompok Blackett terlibat dalam upaya penelitian yang berkaitan dengan pendeteksian kapal dan kapal selam dengan menggunakan radar pesawat terbang. Blackett kemudian memimpin Naval Operational Research pada Angkatan Laut Kerajaan Inggris Raya. Blackett bertugas melakukan studi untuk kepentingan keberhasilan operasi-operasi yang dijalankan oleh Angkatan Laut. Prinsip-prinsip metode ilmiah yang dipakai untuk membantu pengambilan keputusan dalam suatu operasi kegiatan sebagaimana yang dilakukan oleh kelompok Blackett dinamai sebagai Riset Operasi (*Operation Research*).

Ketika Amerika Serikat terlibat dalam Perang Dunia II, prinsip-prinsip riset operasi juga diterapkan, terutama oleh Angkatan Laut dan Angkatan Udara. Peran utama aplikasi riset operasi tampak pada keberhasilan operasi pendaratan

bala tentara Sekutu di Pantai Normandia, Perancis (D Day Operation). Operasi D Day melibatkan 26 kelompok riset operasi. Tiap kelompok riset operasi memiliki 10 orang ilmuwan. Kelompok-kelompok riset operasi tersebut bertugas untuk menganalisis data serangan udara dan laut untuk melawan gempuran dari tentara Nazi Jerman (German U-boats).

Perkembangan riset operasi kemudian menunjukkan perkembangan yang sangat pesat, terutama di Amerika Serikat. Sifat dari industri Amerika yang sangat agresif dalam upayanya meningkatkan produktivitas menyebabkan popularitas riset operasi juga naik. Sifat dunia bisnis yang mirip dengan sifat peperangan memberikan keyakinan bahwa aplikasi riset operasi dalam dunia militer pasti bisa juga diterapkan di dunia bisnis.

Sesaat setelah berakhirnya Perang Dunia II, tahun 1948, pengajaran pertama riset operasi mulai dilakukan di perguruan tinggi, yaitu di Massachusetts Institute of Technology (MIT). Langkah ini kemudian diikuti oleh University of College, London. Bahkan Case Western Reserve, di Amerika Serikat merupakan perguruan tinggi yang kemudian menjadikan riset operasi sebagai suatu program studi tersendiri. Bangunan keilmuan riset operasi semakin kokoh dengan dibentuknya asosiasi keilmuan riset operasi di Amerika Serikat, yaitu Operations Research Society of America. Salah satu kontribusi dari asosiasi ini adalah adanya penerbitan jurnal ilmiah, yaitu Journal of The Operations Research Society of America. Penerbitan jurnal ini kemudian menjadi sarana publikasi dari berbagai hasil penelitian riset operasi di berbagai bidang.

Tujuan utama dari setiap aplikasi riset operasi adalah tercapainya optimasi hasil dari kemungkinan perencanaan yang dibuat. Walaupun pada mulanya riset operasi banyak diaplikasikan untuk kepentingan militer, namun saat ini riset operasi juga mulai banyak sekali diaplikasikan pada berbagai ranah yang lain, seperti ekonomi, bisnis, rekayasa, dan sosial. Berbagai pemanfaatan riset operasi menunjukkan adanya kesamaan karakteristik, yaitu:

1. Pendekatan sistem (*systemics approach*)
2. Pemodelan kuantitatif (*quantitative modelling*)
3. Pendekatan kelompok (*team approach*)

Riset operasi banyak bertautan dengan beberapa bidang ilmu lain. Beberapa pihak bahkan menganggap telah terjadi tumpang-tindih (*overlapping*) antara riset operasi dengan beberapa disiplin ilmu tersebut, seperti manajemen, ilmu komputer, statistik, rekayasa sistem, dan rekayasa industri.

Pemahaman riset operasi sangat dipengaruhi oleh pemahaman terhadap berbagai bidang ilmu yang berkaitan tersebut. Tulisan berikut akan menyajikan beberapa bidang ilmu yang berkaitan dengan pertumbuhan bidang ilmu riset operasi.

1.3 Perkembangan *Operations Research*

Istilah Riset Operasional (*Operation Reseach*) pertama kali digunakan pada tahun 1940 di suatu kota kecil Bowdsey Inggris. Riset Operasi adalah suatu metode pengambilan keputusan yang dikembangkan dari studi operasional-operasional militer selama Perang Dunia II. Dalam perang tersebut, Riset Operasi digunakan dalam mengalokasikan sumber-sumber atau input yang terbatas guna melayani berbagai operasi militer dan kegiatan-kegiatan di dalam setiap operasi secara efisien dan efektif. Tujuannya untuk menerapkan pendekatan ilmiah guna memecahkan permasalahan atau persoalan pada operasi militer ditambah lagi dengan permasalahan strategi dan taktis militer. Setelah bidang militer yang sudah dinyatakan sukses, industri secara bertahap mengaplikasi penggunaan riset operasi, pada tahun 1951 dunia industri dan bisnis dalam riset operasi memberikan dampak besar pada organisasi manajemen.

Perkembangan riset operasi (*Operation Research*) selama Perang Dunia Kedua Di antara sekutu dilakukan oleh James B Connant sebagai pejabat Ketua Komite Rise Pertahanan Nasional, bersama Vannevar Bush sebagai Ketua Komite Persen-jataan dan Perlengkapan Modern Markas Besar Staf Gabungan Angkatan Bersenjata Amerika Serikat. Mereka berdua telah mempelajari dan memantau kelompok Inggris sejak tahun 1940 sampai tahun 1942 dalam perkembangan riset operasi. Setelah keberhasilan penerapan Riset Operas dalam peperangan oleh kalangan militer, dan didorong ole kebangkitan dunia dari kekacauan Perang Dunia Kedua, maka barulah riset operasi sipil menampakkan kemajuan nyata di Amerika Serikat. Atas keberhasilan Riset Operasi dalam perang tersebut para ahli di Inggris dan Amerika, terutama mereka yang bekerja di bidang bisnis, mulai melirik penggunaan metode ini demi keberhasi-lan usaha mereka, dan hasilnya sangat memuaskan.

Dua faktor penting dalam pertumbuhan riset operasi yang maju pada saat ini adalah berkat kemajuan teknik-teknik pemecahan masalah yang berbasis pada kekhasan proses penemuan solusi dan perkembangan *information and communication technology* (ICT) yang membuat waktu hitung lebih efisien dan kualitas akurasi lebih memuaskan. Kemajuan yang terjadi pada awal peningkatan teknik-teknik pemecahan masalah bagi riset operasi, di antaranya adalah metode simpleks yang digunakan untuk menyelesaikan masalah Pemrograman Linier (*Linear Programming*), yang dikembangkan oleh George Dantzing pada tahun 1947. Metode ini sampai sekarang masih banyak digunakan dalam bidang ilmu ekonomi termasuk pertanian, dengan tujuan untuk menentukan nilai optimal dari sumberdaya yang terbatas. Selain itu, berkembang pula teknik model Pemrograman Dinamis (*Dynamic Programming*), Teori Antrian (*Queuing Theory*), Teori Persediaan (*Inventory Theory*), Transportasi (*Transportation*), Perencanaan Jaringan Kerja (*Network Planning*), Teori Permainan (*Game Theory*), yang relatif sudah berkembang dengan baik dari Tahun 1950-an sampai sekarang.

Banyak keputusan utama yang harus diambil oleh manajer dalam perusahaan untuk mencapai tujuan yang diinginkan perusahaan dalam situasi lingkungan yang serba terbatas. Batasan-batasan tersebut meliputi terbatasnya sumber daya seperti tenaga kerja, waktu, bahan baku, uang, dan lain-lain. Riset operasi banyak diterapkan dalam menyelesaikan masalah-masalah manajemen untuk meningkatkan produktivitas dan efisiensi.

Riset operasi dimulai sejak revolusi industri dilakukan. Dunia usaha mengalami perubahan dalam hal ukuran (besarnya) dan kompleksitas organisasi-organisasi perusahaan. Bagian yang mengalami perubahan yang cukup menyolok adalah perkembangan dalam pembagian kerja dan segmentasi tanggung jawab manajemen dalam organisasi-organisasi tersebut. Di sisi lain, organisasi-organisasi (perusahaan) pada saat ini harus beroperasi di dalam situasi dan kondisi lingkungan bisnis yang dinamis dan selalu bergejolak, serta siap untuk berubah-ubah. Perubahan-perubahan tersebut terjadi sebagai akibat dari kemajuan teknologi yang begitu pesat ditambah dengan dampak dari beberapa faktor-faktor lingkungan lainnya seperti keadaan ekonomi, politik, sosial dan sebagainya. Perkembangan kemajuan teknologi tersebut telah menghasilkan dunia komputerisasi. Buah-buah pembangunan telah melahirkan para pimpinan dan pengambilan keputusan, para peneliti, perencana dan pendidik untuk memikirkan serta memecahkan/

menganalisis permasalahan, mengambil langkahlangkah dan strategi yang tepat serta target yang sesuai secara sistematis dalam rangka mencapai tujuan yang telah ditentukan, yakni hasil yang memuaskan. Hasil yang memuaskan tersebut adalah hasil yang optimal yang berarti dampak positifnya maksimum dan dampak negatifnya minimum.

1.4 Definisi *Operations Research* Menurut Para Ahli

Beberapa definisi riset operasi berkenaan dengan pengambilan keputusan yang optimal berdasarkan beberapa orang ahli. Menurut Morse dan Kimbal: Riset Operasi adalah metode ilmiah (*scientific method*) yang memungkinkan para manajer mengambil keputusan mengenai kegiatan yang mereka tangani secara kuantitatif.

Menurut Curchma, Arkoff dan Arnoff: Riset Operasi adalah aplikasi metode-metode, teknik-teknik dan peralatan ilmiah dalam menghadapi masalah yang timbul di dalam operasi perusahaan dengan tujuan ditemukannya pemecahan optimum dari masalah-masalah tersebut.

Menurut Miller dan Starr: Riset Operasi adalah sebagai peralatan manajemen yang menyatukan ilmu pengetahuan, matematika dan logika dalam kerangka pemecahan masalah yang dihadapi sehari-hari, sehingga akhirnya permasalahan tersebut dapat dipecahkan secara optimal.

Dari beberapa definisi tersebut dapat disimpulkan bahwa riset operasi berkenaan dengan pengambilan keputusan yang optimal dalam, dan penyusunan model dari sistem-sistem baik yang deterministik maupun probabilistik yang berasal dari kehidupan nyata. Atau dunia pengelolaan atau dunia usaha yang memakai pendekatan ilmiah atau pendekatan sistematis disebut riset operasi (*Operations Research*).

Tim-tim riset operasi dalam lingkungan dunia bisnis ini menandai kemajuan teknik-teknik riset operasi. Sebagai contoh utama adalah metode simpleks untuk pemecahan masalah-masalah linear programming, yang dikembangkan oleh George Dantzig dalam tahun 1947. Di samping itu banyak peralatan-peralatan riset operasi standar, seperti *linear programming*, *dynamic programming*, teori antrian dan teori pengendalian persediaan telah dikembangkan sebelum akhir tahun 1950-an.

Sehingga Riset operasi berkenaan dengan pengambilan keputusan optimal dan penyusunan model dari sistem-sistem baik deterministik maupun probabilistik yang berasal dari kehidupan nyata. Riset operasi juga mengandung pendekatan atau aplikasi sangat berguna dalam menghadapi masalah bagaimana mengarahkan dan mengkoordinasikan operasi-operasi atau kegiatan-kegiatan dalam suatu organisasi dengan segala batasan-batasannya melalui prosedur “*search for optimality*”, berkenaan dengan penggunaan matematika dan logika dalam pengambilan keputusan operasi sehingga diperoleh hasil yang terbaik.

1.5 Tahapan Studi *Operations Research*

Penyelesaian permasalahan keputusan tidak dapat diselesaikan sendiri oleh seorang ahli Riset Operasional (RO). Permasalahan keputusan diselesaikan oleh tim yang dapat terdiri dari bagian yang mengimplementasikan solusi RO. Tahapan utama dalam studi RO adalah:

1. Identifikasi permasalahan.
2. Pembangunan model.
3. Penyelesaian model.
4. Validasi model.
5. Implementasi hasil akhir.

Kegiatan yang dilakukan pada tahap pertama terdiri dari penentuan tujuan optimasi, identifikasi alternatif keputusan dan sumber daya yang membatasi kegiatan atau aktivitas untuk mencapai tujuan. Tahapan ini akan dilakukan secara bersama-sama antara analis RO dengan pengguna atau pengambil keputusan. Jika identifikasi permasalahan sudah jelas dan lengkap, model keputusan dapat dibangun. Model yang paling tepat harus digunakan, karena kesalahan pembentukan model akan mengakibatkan kesalahan pencapaian solusi optimum. Tahapan ini akan dikerjakan sendiri oleh analis RO. Pemilihan model juga akan didasarkan pada waktu dan biaya yang tersedia.

Tahapan penyelesaian model dilakukan dengan memilih salah satu teknik yang tersedia di RO. Penyelesaian dapat dilakukan menggunakan perangkat lunak komputer karena cukup tersedia perangkat lunak dengan berbagai kemampuan di pasaran. Untuk model yang sederhana tentunya dengan mudah dapat diselesaikan

secara manual dengan atau tanpa bantuan kalkulator. Model dinyatakan valid jika dapat memberikan prediksi yang masuk akal akan kinerja sistem. Metode umum yang digunakan untuk memeriksa validitas model adalah membandingkan solusi yang diperoleh dengan data lalu yang tersedia dari sistem nyata. Model dikatakan valid jika pada kondisi input yang sama dengan sistem nyata menghasilkan kinerja sistem yang sama dengan sistem nyata.

Tahap terakhir merupakan implementasi. Tahapan ini mencakup penerjemahan solusi optimal yang diperoleh pada tahap penyelesaian model ke dalam instruksi operasional yang dapat dimengerti oleh individu yang menjalankan sistem.

1.6 Dampak Operations Research

Beberapa jenis persoalan yang telah dipecahkan dengan menggunakan teknik-teknik dalam RO antara lain *linear programming*, *dynamic programming*, teori antrian, teori inventori, teori permainan (*game theory*), simulasi, network planning. Linnear progamming telah dipergunakan dan telah mencapai sukses di dalam pemecahan persoalan yang berkenaan dengan penugasan personel (*assignment of personnel*), *blending of materials*, distribusi dan transportasi serta *Investment*. *Dynamic programming* telah berhasil diterapkan dalam perencanaan pengeluaran periklanan, usaha mendistribusikan penjualan dan penjadwalan produksi (*production scheduling*). Teori antrian (*queuing or waiting line theory*) berhasil diterapkan dalam memecahkan kemacetan lalu lintas (*traffic congestion*), pelayanan mesin-mesin akibat perusakan, penentuan jumlah pemberi pelayanan yang optimal, penjadwalan lalu lintas udara (*air traffic scheduling*), mendesain dam, penjadwalan produksi, meminimumkan waktu menunggu untuk menerima pelayanan, operasi dalam rumah sakit, dan lain sebagainya. Teknik-teknik RO lainnya seperti teori inventori, teori permainan (*game theory*) dan simulasi telah menunjukkan sukses yang besar dalam pemecahan beberapa jenis permasalahan/persoalan.

1.7 Model Manfaat *Operations Research*

1.7.1 Manfaat *Operations Research*

1. Merupakan alat untuk pengambilan keputusan yang optimal dari berbagai sumber daya yang tersedia.
2. Riset Operasi berusaha menetapkan arah tindakan terbaik (optimum) dari sebuah keputusan masalah dengan pembatasan sumber daya terbatas.
3. Memberikan pengembangan dari beberapa sektor keilmuan, seperti matematik, teknik dan ilmu perhitungan, ilmu politik, ekonomi, teori probabilitas dan statistik.
4. Memberikan kemudahan dalam pengambilan keputusan kegiatan kerja dalam bidang industri, bisnis, dan manajemen.

1.7.2 Model *Operations Research*

Abstraksi atau penyederhanaan realita sistem yang kompleks dimana hanya komponen-komponan yang relevan atau faktor-faktor yang dominan dari masalah yang dianalisis dan diikutsertakan. Banyak *Model Operations Research* yang sudah dikembangkan dan digunakan terhadap persoalan-persoalan bidang usaha. Model tersebut dapat dikelompokkan ke dalam beberapa jenis, yaitu:

1. **Linear Programming**

Program ini memuat metode grafik, simpleks, dan dualitas yang digunakan pada proses alokasi. Program ini akan menjawab persoalan bila:

Terdapat sejumlah kegiatan untuk dilaksanakan dan terdapat alternative cara untuk melaksanakannya. Sumber dan fasilitas tidak tersedia untuk melaksanakan tiap kegiatan dengan cara yang paling efektif. Persoalan ialah menggabungkan kegiatan dan sumber sedemikian rupa hingga terdapat efektivitas keseluruhan secara maksimal.

2. **Metode Transportasi**

Persoalan ini merupakan bagian khusus dari proses alokasi. Metode ini mempunyai cara tersendiri untuk menjawab persoalan alokasi seperti cara bantu loncatan (metode *stepping stone*), cara MODI, dan cara pendekatan Vogel's.

3. **Metode Penugasan**

Model ini berhubungan dengan penugasan optimal dari bermacam-macam sumber yang produktif atau personalia yang mempunyai tingkat efisiensi yang berbeda-beda untuk tugas-tugas yang berbeda pula.

4. **Teori Network**

Memuat persoalan-persoalan serta pemecahan dari proyek manajemen yang menyangkut perencanaan serta penjadwalan. Alat yang digunakan adalah CPM dan PERT.

5. **Teori Keputusan**

Ciri penting dari teori keputusan adalah bahwa akibat dari tindakan, umumnya tidak diketahui. Dalam hal ini, peluang dihubungkan dengan bermacam-macam keadaan. Kita dapat menunjuk keputusan tentang kepastian, resiko dan ketidakpastian, tergantung pada seberapa banyak kita mengetahui keadaan. Cara lain untuk menaksir masa depan meski hanya tersedia sejumlah kecil informasi adalah dengan statistik Bayes.

6. **Teori Permainan**

Teori permainan ini memberikan rangka konsepsi dalam mana persoalan kompetisi dapat dirumuskan. Teori ini telah dipergunakan secara efektif oleh dunia usaha untuk mengembangkan strategi periklanan, kebijakan harga, dan waktu pengenalan produksi baru.

7. **Model-model Ikonis / Fisik**

Penggambaran fisik dari suatu sistem, baik dalam bentuk ideal maupun dalam skala yang berbeda. Contoh: foto, peta, mainan anak-anak, maket, histogram.

8. **Model Analog / Diagramatis**

Model-model ini dapat menggambarkan situasi-situasi yang dinamis, dan lebih banyak digunakan daripada model-model ikonis karena sifatnya yang dapat dijadikan analogi bagi karakteristik sesuatu yang dipelajari. Contoh: kurva distribusi frekuensi pada statistik, flow chart, peta dengan bermacam-macam warna untuk menggambarkan kondisi sebenarnya.

9. **Model Simbolis / Matematika**

Penggambaran dunia nyata melalui simbol-simbol matematis. Model ini menggunakan seperangkat simbol matematik untuk menunjukkan komponen-komponen dari sistem nyata. Namun demikian, sistem nyata tidak selalu dapat diekspresikan dalam rumusan matematik. Contoh: persamaan garis lurus $y = ax + b$; $z = x_1 + x_2 + x_3$. Model matematik dapat dibedakan menjadi dua kelompok, yaitu: deterministik dan probabilistik. Model deterministik dibentuk dalam situasi penuh kepastian, sedangkan model probabilistik meliputi kasus-kasus dimana diasumsikan penuh ketidakpastian.

10. **Model Simulasi**

Model-model yang meniru tingkah laku sistem dengan mempelajari interaksi komponen-komponennya. Karena tidak memerlukan fungsifungsi matematis secara eksplisit untuk merealisasikan variabel-variabel sistem, maka model-model simulasi ini dapat digunakan untuk memecahkan sistem kompleks yang tidak dapat diselesaikan secara matematis. Akan tetapi, model-model ini tidak dapat memberikan solusi yang benar-benar optimum.

11. **Model Heuristik**

Kadang-kadang formulasi matematis bersifat sangat kompleks untuk dapat memberikan suatu solusi yang pasti, atau mungkin suatu solusi optimum dapat diperoleh, akan tetapi memerlukan proses perhitungan yang sangat panjang dan tidak praktis. Untuk mengatasi kasus seperti ini dapat digunakan metode heuristik, yaitu suatu metode pencarian yang didasarkan atas intuisi atau aturan-aturan empiris untuk memperoleh solusi yang lebih baik daripada solusi-solusi yang telah dipelajari sebelumnya.

1.8 Langkah dan Tujuan *Operations Research*

1.8.1 Langkah *Operations Research*

Langkah-langkah yang dikemukakan oleh Borg & Gall sangat bersesuaian dengan langkah-langkah yang dikemukakan oleh Kirk Patrick. Karena itulah peneliti menyederhanakan langkah-langkah tersebut. Prosedur yang dilakukan dalam penelitian pengembangan ini mengkombinasikan langkah-langkah yang dikemukakan oleh Borg & Gall dengan prosedur pengembangan dalam model Kirk Patrick melalui empat tahap, yaitu:

1. Tahap Awal
2. Tahap Desain
3. Tahap Uji Coba dan Revisi
4. Tahap implementasi

1.8.2 Tujuan *Operations Research*

Tujuan dari Riset Operasi adalah menerapkan pendekatan ilmiah guna memecahkan permasalahan atau persoalan memikirkan serta memecahkan/menganalisis permasalahan, mengambil langkah-langkah dan strategi yang tepat serta target yang sesuai secara sistematis dalam rangka mencapai tujuan yang telah ditentukan, yakni hasil yang memuaskan. Hasil yang memuaskan tersebut adalah hasil yang optimal yang berarti dampak positifnya maksimum dan dampak negatifnya minimum.

1.9 Penerapan dalam Kehidupan Sehari-hari

Sejalan dengan perkembangan dunia industri dan didukung dengan kemajuan di bidang komputer, Riset Operasi semakin diterapkan di berbagai bidang untuk menangani masalah yang cukup kompleks. Berikut ini adalah contoh-contoh penggunaan Riset Operasi di beberapa bidang di antaranya:

- a) Akuntansi dan Keuangan.
- b) Penentuan jumlah kelayakan kredit.
- c) Alokasi modal investasi dari berbagai alternatif.
- d) Peningkatan efektivitas akuntansi biaya.
- e) Penugasan tim audit secara efektif.

Pemasaran:

1. Penentuan kombinasi produk terbaik berdasarkan permintaan pasar.
2. Alokasi iklan di berbagai media.
3. Penugasan tenaga penjual kewilayah pemasaran secara efektif..
4. Penempatan lokasi gudang untuk meminimumkan biaya distribusi.
5. Evaluasi kekuatan pasar dari strategi pemasaran pesaing

Operasi Produksi:

1. Penentuan bahan baku yang paling ekonomis untuk kebutuhan pelanggan.
2. Meminimumkan persediaan atau inventori.
3. Penyeimbangan jalur perakitan dengan berbagai jenis operasi..
4. Peningkatan kualitas operasi manufaktur.
5. Aplikasi riset operasi juga mempunyai dampak yang kuat dalam studi masalah sosial dan pekerjaan umum. Orang menjadi lebih sadar tentang bagaimana riset operasi dapat membantu aktivitas pembagian keputusan sehari-hari.

1.10 Metodologi *Operation Research*

Pembentukan model yang cocok hanyalah salah satu tahap dari aplikasi RO. Pola dasar penerapan RO terhadap suatu masalah dapat dipisahkan menjadi beberapa tahap. Berikut adalah langkah-langkah (metodologi) untuk memecahkan persoalan dalam organisasi.

1. Merumuskan Masalah

Sebelum solusi terhadap suatu permasalahan dipikirkan, pertama kali yang harus dilakukan adalah mendefinisikan atau merumuskan permasalahan dengan baik. Definisi masalah yang tidak baik akan menyebabkan tidak diperoleh penyelesaian atas suatu masalah atau penyelesaian yang tidak tepat. Dalam perumusan masalah ini ada tiga pertanyaan penting yang harus dijawab:

- a) Variabel keputusan, yaitu unsur-unsur dalam persoalan yang dapat dikendalikan oleh pengambil keputusan. Ia sering disebut sebagai instrumen.

- b| Tujuan. Penetapan tujuan membantu pengambil keputusan memusatkan perhatian pada persoalan dan pengaruhnya terhadap organisasi. Tujuan ini diekspresikan dalam variabel keputusan.
- c| Kendala adalah pembatas-pembatas terhadap alternatif tindakan yang tersedia.

2. Pembentukan Model

Sesuai dengan definisi permasalahannya, kelompok peneliti RO tersebut harus menentukan model yang paling cocok untuk mewakili sistem yang bersangkutan. Model tersebut harus merupakan ekspresi kuantitatif dari tujuan dan batasan-batasan persoalan dalam bentuk variabel keputusan. Dalam memformulasikan permasalahan, biasanya digunakan model analitik, yaitu model matematik yang menghasilkan persamaan. Jika pada suatu situasi yang sangat rumit tidak diperoleh model analitik, maka perlu dikembangkan suatu model simulasi.

3. Pemecahan Model

Pada tahap ini, bermacam-macam teknik dan metode solusi kuantitatif yang merupakan bagian utama dari RO memasuki proses. Penyelesaian masalah sesungguhnya merupakan penerapan satu atau lebih teknikteknik ini terhadap model. Seringkali, solusi terhadap model berarti nilainilai variable keputusan yang mengoptimumkan salah satu fungsi tujuan dengan nilai fungsi tujuan lain yang dapat diterima. Di samping solusi model, perlu juga mendapat informasi tambahan mengenai tingkah laku solusi yang disebabkan karena perubahan parameter sistem. Ini biasanya dinamakan sebagai Analisis Sensitivitas. Analisis ini terutama diperlukan jika parameter sistem tak dapat diduga secara tepat.

4. Validasi Model

Sebuah model adalah absah jika, walaupun tidak secara pasti mewakili system tersebut, dan dapat memberikan prediksi yang wajar dari kinerja system tersebut. Suatu metode yang biasa digunakan untuk menguji validitas model adalah dengan membandingkan kinerjanya dengan data masa lalu yang tersedia. Model dikatakan valid jika dengan kondisi input yang serupa dapat menghasilkan kembali kinerja seperti masa lampau. Masalahnya adalah bahwa tidak ada yang menjamin kinerja masa depan akan berlanjut meniru cerita lama.

5. Implementasi Akhir

Tahap terakhir adalah menerapkan hasil model yang telah diuji. Hal ini membutuhkan suatu penjelasan yang hati-hati tentang solusi yang digunakan dan hubungannya dengan realitas. Suatu hal yang kritis pada tahap ini adalah mempertemukan ahli RO dengan mereka yang bertanggung jawab terhadap pelaksanaan sistem. Penyelesaian kelima langkah yang dijelaskan di atas bukan berarti proses ini telah selesai. Hasil model dan keputusan hasil yang tersedia memberikan umpan balik pada model awal.

1.11 Soal dan Kasus

Soal:

Telitilah mana di antara model-model berikut ini yang dapat diselesaikan dengan program linier:

Maksimumkan $f() = 5 + x_2^2$ Kendala:

$$- + 4 = 3$$

$$3 + 4x_2 = -5$$

$$, \geq 0$$

Minimumkan f , = 5 + $2x_2$ -

Kendala:

$$2 \quad 4 \quad = 3$$

$$5 \quad + x_2 - 2x_3 = 2$$

$$, \geq 0$$

Minimumkan f , = Kendala:

$$4 \quad + \quad = 2$$

$$3 \quad = 1$$

Maksimumkan f , = + Kendala:

$$+ 2x_2- \geq 1$$

$$3x_1 + 2x_2 \leq 2$$

$$x_1, x_2 \geq 0$$

Minimumkan $f(x_1, x_2) = x_1^2 x_2^2$ Kendala:

$$\geq e^3$$

$$x_1 x_2^4 \geq e^4 x_1^2 x_2^3 \geq$$

$$x_1, x_2 \geq 0$$

Sebuah perusahaan mebel akan membuat meja dan kursi. Setiap meja membutuhkan 5 m² kayu jati dan 2 m² kayu pinus, serta membutuhkan waktu pembuatan selama 4 jam. Untuk membuat sebuah kursi dibutuhkan 2 m² kayu jati, 3 m² kayu pinus dan 2 jam kerja. Dari penjualan sebuah meja didapatkan keuntungan sebesar Rp12.000,00 sedangkan keuntungan dari sebuah kursi adalah Rp8000,00. Mebel itu ingin membuat sebanyak-banyaknya, tetap terbatas dalam bahan baku dan tenaga kerja. Dalam seminggu hanya mampu mendapatkan 150 m² kayu jati, 100 m² kayu pinus, serta hanya memiliki 80 jam kerja.

Masalah:

Berapa buah meja dan kursi yang harus ia buat mengingat kendala yang ada, supaya ia memperoleh keuntungan sebanyak-banyaknya? Buatlah model riset operasi (variabel keputusan, tujuan dan kendala) dari permasalahan di atas disertakan penjelasannya!

Penyelesaian:

Model-model berikut ini yang dapat diselesaikan dengan program linier:

Bukan merupakan bentuk program linier karena fungsinya mengandung suku $x^{\frac{2}{2}}$, yang jelas bukan merupakan bentuk linier.

Bukan merupakan bentuk program linier meskipun fungsinya merupakan bentuk linier dalam x_1 dan x_2 , tetapi ada kendala yang memuat bentuk perkalian variabel

(4) . Perhatikan di sini bahwa meskipun fungsi sasaran

maupun kendala lain sudah berbentuk linier, namun jika ada satu kendala saja yang tidak berbentuk linier, maka model tidak bias diselesaikan dengan program linier.

Model program linier. Tampak bahwa baik fungsi maupun kedua kendala merupakan bentuk fungsi linier dalam dan Meskipun tidak ada syarat ≥ 0 , dengan sedikit transformasi, bentuk tetap dapat diselesaikan dengan program linier.

Model program linier dalam 3 variabel, dan .

Meskipun kendala berbentuk pertidaksamaan, tetapi dengan transformasi sederhana dapat dijadikan ke bentuk persamaan (cara transformasi dibahas dalam Bab 3). Perhatikan juga bahwa meskipun merupakan model dalam 3 variabel, dan, tetapi tidak semua variabelnya muncul dalam fungsi sasaran maupun kendalanya. Fungsi sasaran $f = 3x_2$ yang merupakan fungsi 2 variabel sama dengan fungsi $f = 3x_2 + 0x_3$ yang merupakan fungsi 3 variabel. Meskipun tampak bahwa model bukan merupakan model program linier, tetapi dengan suatu transformasi dapat dijadikan program linier. Fungsi $\ln(x)$ merupakan fungsi monoton sehingga meminimumkan $f(x)$ sama dengan meminimumkan $\ln(f(x))$. Misalkan $y_1 = \ln(x_1)$, $y_2 = \ln(x_2)$. Dengan mengingat bahwa $\ln(xy) = \ln(x) + \ln(y)$, $\ln(e) = 1$, $\ln(a)$ dan $\ln(e) = 1$, fungsi sasaran dapat dituliskan sebagai:

$$\begin{aligned} g \ y_1 y_2 &= \ln f \ x \frac{2}{1} x \frac{2}{2} \\ &= \ln x \frac{2}{1} + \ln x \frac{2}{2} \\ &= 2 \ln x_1 + 2 \ln x_2 \\ &= 2y_1 + 2y_2 \end{aligned}$$

Maka model hasil transformasi adalah:

Minimumkan $g \ y_1 y_2 = 2y_1 + 2y_2$

Kendala:

$$\begin{aligned} y_1 + 2y_2 &\geq 3 \\ y_1 + 4y_2 &\geq 4 \\ 2y_1 + 3y_2 &\geq 3 \\ y_1, y_2 &\geq 0 \end{aligned}$$

Penyelesaian:

Keuntungan ditentukan oleh seberapa banyak meja dan kursi yang dibuat. Oleh karena itu dibuat variabel keputusan sebagai berikut:

Misalkan:

x_1 = Jumlah meja yang harus dibuat

x_2 = Jumlah kursi yang harus dibuat

Tujuan:

Tujuan dari perusahaan tersebut adalah memaksimalkan keuntungan. Keuntungan sebuah meja adalah Rp12.000,00 dan sebuah kursi adalah Rp8.000,00. Karena ia membuat meja dan kursi (x_1 dan x_2 adalah besaran yang akan dicari), maka total keuntungan yang ia peroleh adalah sebesar:

$$f(x_1, x_2) = 12.000x_1 + 8.000x_2$$

Fungsi inilah yang akan dioptimalkan (dalam kasus ini dimaksimalkan). Jika tidak ada kendala, penyelesaian masalah ini menjadi mudah, yaitu dengan membuat dan sebesar-besarnya. Dengan memperbanyak jumlah meja dan kursi yang dibuat, maka perusahaan itu akan memperoleh keuntungan yang semakin besar. Tetapi keadaan itu tidak dapat dicapai mengingat keterbatasan bahan baku (kayu jati dan pinus) serta tenaga kerja.

Kendala:

Keterbatasan bahan baku dan tenaga kerja dapat dinyatakan dalam tabel di bawah ini:

Sumber Daya	Meja	Kursi	Persediaan
Kayu Jati	5	2	150
Kayu Pinus	2	3	100
Jam Kerja	4	2	80

Dengan membuat x_1 buah meja dan x_2 buah kursi, maka kendala yang harus dipenuhi adalah:

$$5x_1 + 2x_2 \leq 150$$

$$2x_1 + 3x_2 \leq 100$$

$$4x_1 + 2x_2 \leq 80$$

$$x_1, x_2 \geq 0$$

(bisa juga ditambahkan syarat bahwa dan bilangan bulat) Dengan demikian model yang sesuai untuk kasus perusahaan mebel di atas adalah:

Maksimumkan $f(x_1, x_2) = 12.000x_1 + 8.000x_2$

Kendala:

$$5x_1 + 2x_2 \leq 150$$

$$2x_1 + 3x_2 \leq 100$$

$$4x_1 + 2x_2 \leq 80$$

$$x_1, x_2 \geq 0$$

Bab 2

Linear Programming

2.1 Pendahuluan Pengertian, Penjelasan dan Contoh

Program linier (linear programming yang disingkat LP) mungkin merupakan salah satu teknik OR yang digunakan paling luas dan diketahui dengan baik. Ia merupakan metode matematika dalam mengalokasikan sumber daya yang langka untuk mencapai tujuan tunggal seperti memaksimalkan keuntungan atau meminimumkan biaya. LP banyak diterapkan dalam membantu menyelesaikan masalah ekonomi, industri, militer, sosial dan lain-lain. LP berkaitan dengan penjelasan suatu dunia nyata sebagai suatu model matematika yang terdiri atas sebuah fungsi tujuan linier dan sistem kendala linier.

Program Linear (Linear Programming) merupakan sebuah teknik matematik yang didesain untuk membantu para manajer operasi dalam merencanakan dan membuat keputusan yang diperlukan untuk mengalokasikan sumber daya berdasarkan pendapat Heizer dan Render (2006).

Kemudian program linier segera ditetapkan dalam bidang pemerintahan dan bisnis. Hasilnya, LP disadari sebagai pendekatan penyelesaian masalah yang sangat ampuh untuk analisis keputusan dalam bidang bisnis. Di samping itu, analisis Input. Output dari Wassily Leontief memberikan suatu dasar untuk menerapkan LP pada analisis ekonomi antar industri. Akhir-akhir ini aplikasi LP telah meningkat dengan perkembangan yang cepat karena dukungan komputer elektronik.

2.1.1 Tujuan dari penggunaan Program Linear

- Mempelajari program linear sebagai penunjang pengambilan keputusan.
 - Memahami syarat-syarat pemecahan program linear dan pemecahannya.
-

- Memperkenalkan metode grafik untuk pemecahan maksimisasi dan minimisasi persoalan program linear.
- Mempelajari masalah teknik dalam program linear seperti titik ekstrim dan alternatif pemecahan optimum dan memperlihatkannya dengan metode grafik.

2.1.2 Syarat pembentukan Model Program Linear

Berikut adalah syarat pembentukan model program linear:

- Variabel keputusan merupakan unsur-unsur dalam persoalan yang dapat dikendalikan oleh pengambil keputusan.
- Persoalan Linear Programming bertujuan untuk memaksimalkan atau meminimalkan kuantitas (pada umumnya berupa laba atau biaya). Sifat umum ini disebut sebagai fungsi tujuan (*objective function*) dari suatu persoalan Linear Programming. Tujuan utama suatu perusahaan pada umumnya untuk memaksimalkan keuntungan pada jangka panjang. Dalam kasus sistem distribusi suatu perusahaan angkutan atau penerbangan, tujuan pada umumnya berupa meminimalkan biaya.
- Adanya batasan (*constraints*) atau kendala, yang membatasi tingkat sampai di mana sasaran dapat dicapai. Sebagai contoh, keputusan untuk memproduksi berapa banyak unit dari tiap produk dalam suatu lini produk perusahaan, dibatasi oleh tenaga kerja dan mesin yang tersedia. Oleh karena itu, untuk memaksimalkan atau meminimalkan suatu kuantitas (fungsi tujuan) bergantung kepada sumber daya yang jumlahnya terbatas (batasan).
- Harus ada beberapa alternatif tindakan yang dapat diambil. Sebagai contoh, jika suatu perusahaan menghasilkan tiga produk berbeda, manajemen dapat menggunakan Linear Programming untuk memutuskan bagaimana cara mengalokasikan sumber dayanya yang terbatas (tenaga kerja, permesinan, dan seterusnya). Jika tidak ada alternatif yang dapat diambil, maka Linear Programming tidak diperlukan.
- Kita harus dapat menyatakan tujuan perusahaan dan segenap keterbatasannya sebagai kesamaan atau ketidaksamaan matematik, dan harus ada kesamaan dan ketidaksamaan linear. Tujuan perusahaan yakni keuntungan.

2.2 Bentuk Umum Program Linear

2.2.1 Formulasi Model LP

Masalah keputusan yang sering dihadapi analis adalah alokasi optimum sumber daya yang langka. Sumber daya dapat berupa uang, tenaga kerja, bahan mentah, kapasitas mesin, waktu, ruangan atau teknologi. Tugas analis adalah mencapai hasil terbaik yang mungkin dengan keterbatasan sumber daya itu. Hasil yang diinginkan mungkin ditunjukkan sebagai maksimisasi dari beberapa ukuran seperti profit, penjualan dan kesejahteraan, atau minimisasi seperti pada biaya, waktu dan jarak.

Setelah masalah diidentifikasi, tujuan ditetapkan, langkah selanjutnya adalah formulasi model matematika yang meliputi tiga tahap seperti berikut:

- Tentukan variabel yang tak diketahui (variabel keputusan) dan nyatakan dalam simbol matematika.
- Membentuk fungsi tujuan yang ditunjukkan sebagai suatu hubungan linier (bukan perkalian) dari variabel keputusan.
- Menentukan semua kendala masalah tersebut dan mengekspresikan dalam persamaan atau pertidaksamaan yang juga merupakan hubungan linier dari variabel keputusan yang mencerminkan, keterbatasan sumber daya masalah itu.

Ingat bahwa pembentukan model bukan bersifat ilmiah murni tetapi lebih bersifat seni dan akan menjadi dimengerti terutama karena praktik. Karena itu pada uraian berikut ini akan disajikan beberapa contoh yang menunjukkan langkah-langkah formulasi model LP.

Siswanto (2007: 26) menyebutkan definisi pemrograman linear yaitu sebagai metode matematis yang berbentuk linear untuk menentukan suatu penyelesaian optimal dengan cara memaksimalkan atau meminimumkan fungsi tujuan terhadap suatu susunan kendala. Secara keseluruhan, berdasarkan definisi maka tujuan pemrograman linear adalah memecahkan persoalan memaksimalkan atau meminimumkan untuk mendapatkan penyelesaian yang optimal.

Model Pemrograman Linear (MPL) memiliki sebuah fungsi objektif dan satu atau lebih kendala. Pada fungsi objektif terdapat parameter yang disebut koefisien fungsi objektif (*objective function coefficients*). Koefisien fungsi objektif menggambarkan kontribusi satu satuan variabel keputusan terhadap nilai fungsi objektif. Koefisien fungsi objektif yang selama ini dikenal dalam pembahasan MPL bersifat tegas, demikian pula dengan kendala.

Fungsi Tujuan merupakan suatu pernyataan matematik dalam pemrograman linear yang memaksimalkan atau meminimalkan kuantitas (sering berupa laba atau biaya, tetapi setiap tujuan dapat digunakan). Sedangkan batasan merupakan pembatas yang membatasi tingkat sampai dimana seorang manajer dapat mencapai suatu tujuan.

Terdapat tiga unsur utama yang membangun suatu program linear yaitu (Siswanto, 2007: 26):

1. Variabel keputusan

Variabel keputusan adalah variabel yang mempengaruhi nilai tujuan yang hendak dicapai. Pada proses pembentukan suatu model, menentukan variabel keputusan merupakan langkah pertama sebelum menentukan fungsi tujuan dan fungsi kendala.

2. Fungsi tujuan

Fungsi tujuan pada model pemrograman linear haruslah berbentuk linear.

Selanjutnya, fungsi tujuan tersebut dimaksimalkan atau diminimalkan terhadap fungsi-fungsi kendala yang ada.

3. Fungsi kendala

Fungsi kendala adalah suatu kendala yang dapat dikatakan sebagai suatu pembatas terhadap variabel-variabel keputusan yang dibuat. Fungsi kendala untuk model pemrograman linear juga harus berupa fungsi linear.

4. Fungsi non-negative

Fungsi yang menyatakan bahwa setiap variabel yang terdapat di dalam model pemrograman linear tidak boleh negatif. Secara matematis ditulis sebagai $X_1, X_2, \dots, X_j \geq 0$.

2.2.2 Asumsi-asumsi Dasar Pemrograman Linear

Asumsi-asumsi dasar pemrograman linear diuraikan agar penggunaan teknik pemrograman linear ini dapat memuaskan untuk berbagai masalah. Asumsi-asumsi dalam pemrograman linear akan dijelaskan secara implisit pada bentuk umum model pemrograman linear. Adapun asumsi-asumsi dasar pemrograman linear sebagai berikut:

a) Kesebandingan

Asumsi ini mempunyai arti bahwa naik turunnya nilai fungsi tujuan dan penggunaan sumber atau fasilitas yang tersedia akan berubah secara sebanding (proportional) dengan perubahan tingkat kegiatan.

b) Penambahan

Asumsi ini mempunyai arti bahwa nilai fungsi tujuan tiap kegiatan tidak saling mempengaruhi, atau dalam pemrograman linear dianggap bahwa kenaikan dari nilai tujuan yang diakibatkan oleh kenaikan suatu kegiatan dapat ditambahkan tanpa mempengaruhi bagian nilai tujuan yang diperoleh dari kegiatan lain.

c) Dapat Dibagi

Asumsi ini menyatakan bahwa keluaran (*output*) yang dihasilkan oleh setiap kegiatan dapat berupa bilangan pecahan. Demikian pula dengan nilai tujuan yang dihasilkan.

d) Kepastian

Asumsi ini menyatakan bahwa semua parameter yang terdapat dalam model pemrograman linear (a_{ij} , b_i , a_j) dapat diperkirakan dengan pasti.

2.2.3 Kesamaan dan Ketidaksamaan Matematik dalam Program Linear

Meskipun kesamaan lebih populer dibandingkan dengan ketidaksamaan namun ketidaksamaan merupakan suatu hubungan yang penting dalam program linear. Kesamaan digambarkan oleh tanda "=" dan merupakan pernyataan khusus dalam matematik.

Namun banyak persoalan perusahaan yang tidak dapat dinyatakan dalam bentuk kesamaan yang jelas dan rapi. Hitungan yang dicari tidak selalu satuan bulat tetapi bisa juga berupa angka kira-kira. Untuk itu dibutuhkan ketidaksamaan yakni hubungan lain yang dinyatakan dalam bentuk matematik. Sebagian besar batasan dalam persoalan program linear dinyatakan sebagai ketidaksamaan.

2.2.4 Bentuk Umum Model Pemrograman Linear

Bentuk umum model Lp itu adalah:

$$\text{Maksimumkan (minimumkan) } Z = \sum_{j=1}^n C_j X_j$$

Dengan syarat:

$$a_{ij} X_j (\leq, =, \geq) b_i, \text{ untuk semua } i(i = 1, 2, \dots, m) \text{ semua } X_j \geq 0$$

keterangan:

X_j : banyaknya kegiatan j , di mana $j = 1, 2, \dots, n$. berarti di sini terdapat n variabel keputusan.

Z : nilai fungsi tujuan

C_j : sumbangan perunit kegiatan, untuk masalah maksimisasi c , menunjukkan keuntungan atau penerimaan per unit, sementara dalam kasus minimisasi ia menunjukkan biaya per unit.

Masalah pemrograman linear adalah masalah optimisasi bersyarat yakni pencarian nilai maksimum atau pencarian nilai minimum sesuatu fungsi tujuan berkenaan dengan keterbatasan-keterbatasan atau kendala yang harus dipenuhi.

Masalah-masalah tersebut secara umum dapat dirumuskan sebagai berikut (Johannes Supranto, 1991: 44):

Fungsi tujuan memaksimumkan dinotasikan dengan Z dan relasi dalam kendala berbentuk (\leq) sehingga bentuknya dapat dilihat pada persamaan (2.1).

Maksimumkan fungsi tujuan

$$Z = c_1X_1 + c_2X_2 + \dots + c_jX_j \quad (2.1)$$

terhadap kendala-kendala

$$\begin{aligned} a_{11}X_1 + a_{12}X_2 + \dots + a_{1j}X_j &\leq b_1 \\ a_{21}X_1 + a_{22}X_2 + \dots + a_{2j}X_j &\leq b_2 \\ a_{i1}X_1 + a_{i2}X_2 + \dots + a_{ij}X_j &\leq b_i \\ X_j &\geq 0 \quad (j = 1, 2, \dots, n) \end{aligned} \quad (2.2) \text{ Kendala non negatif}$$

Fungsi tujuan meminimumkan dinotasikan dengan W dan relasi dalam kendala berbentuk (\geq) sehingga menjadi:

Meminimumkan fungsi tujuan $W = c_1X_1 + c_2X_2 + \dots + c_jX_j$

terhadap kendala-kendala

$$\begin{aligned} a_{11}X_1 + a_{12}X_2 + \dots + a_{1j}X_j &\geq b_1 \\ a_{21}X_1 + a_{22}X_2 + \dots + a_{2j}X_j &\geq b_2 \\ a_{i1}X_1 + a_{i2}X_2 + \dots + a_{ij}X_j &\geq b_i \\ X_j &\geq 0 \quad (j = 1, 2, \dots, n) \end{aligned} \quad \text{Kendala non negatif}$$

dengan:

- X_j : variabel keputusan ke- j /banyaknya produk ke- j ($j = 1, 2, \dots, n$)
- b_i : suku tetap/bahan mentah jenis ke- i yang tersedia ($i = 1, 2, \dots, m$)
- a_{ij} : koefisien kendala/bahan mentah ke- i yang digunakan untuk memproduksi satu unit produk j
- c_j : koefisien ongkos/harga jual satu unit j

2.2.5 Contoh Formulasi Model LP

Contoh 1: Masalah Kombinasi Produk

Sebuah perusahaan menghasilkan tiga jenis produk, yaitu sepatu, tas dan dompet. Jumlah waktu kerja buruh yang tersedia adalah 240 jam kerja dan bahan mentah 400 Kg dan harga masing-masing produk adalah seperti yang tersaji pada Tabel 2.1. Apa yang harus dilakukan perusahaan ini?

Tabel 2.1

Jenis Produk	Kebutuhan Sumber daya		Harga (Rp/Unit)
	Buruh (Jam/unit)	Bahan (Kg/unit)	
Produk 1 (Sepatu)	5	4	3
Produk 2 (Tas)	2	6	5
Produk 3 (Dompet)	4	3	2

Pada kasus ini, masalah yang dihadapi perusahaan adalah menentukan jumlah masing- masing produk yang harus dihasilkan agar keuntungan maksimum. Sekarang kita akan merumuskan masalah dalam suatu model LP.

a) Variabel keputusan

Tiga variabel dalam masalah ini adalah jumlah sepatu, tas, dan dompet yang harus dihasilkan. Jumlah ini dapat dilambangkan sebagai:

X_1 = jumlah produk 1

X_2 = jumlah produk 2

X_3 = jumlah produk 3

b) Fungsi tujuan

Tujuan dari masalah kombinasi produk adalah untuk memaksimalkan penerimaan total. Jelas bahwa penerimaan total adalah jumlah penerimaan yang diperoleh dari masing-masing produk. Penerimaan dari produk 1 adalah perkalian antara jumlah produk 1 dengan harga per unit (Rp3). Penerimaan produk 2 dan 3 ditentukan dengan cara serupa. Sehingga penerimaan total, Z , dituliskan sebagai:

$$Z = 3X_1 + 5X_2 + 2X_3$$

c) **Sistem kendala**

Dalam masalah ini kendalanya adalah jumlah buruh dan bahan mentah yang terbatas. Masing-masing produk membutuhkan baik buruh maupun bahan mentah. Bagi produk 1, buruh yang dibutuhkan untuk menghasilkan tiap unit adalah 5 jam, sehingga buruh yang dibutuhkan untuk produk 1 adalah $5X_1$ jam. Dengan cara serupa, produk 2 membutuhkan $2X_2$ jam buruh, dan produk 3 membutuhkan $4X_3$ jam. Jumlah jam buruh yang tersedia adalah 240. Sehingga, kendala buruh dituliskan:

$$5X_1 + 2X_2 + 4X_3 \leq 240$$

Kendala bahan mentah dirumuskan dengan cara yang sama. Produk 1 membutuhkan 4 Kg per unit, produk 2 membutuhkan 6 Kg per unit, dan produk 3 membutuhkan 6 Kg per unit. Karena terdapat 400 Kg bahan mentah maka kendala ini dituliskan sebagai:

$$4X_1 + 6X_2 + 3X_3 \leq 400$$

Kita juga membatasi masing-masing variabel hanya pada nilai positif, karena akan tidak masuk akal untuk menghasilkan jumlah produk negatif. Kendala-kendala ini dinamakan *non-negativity constraints* dan secara matematika dituliskan sebagai:

$$X_1 > 0, X_2 > 0, X_3 \geq 0$$

Hampir semua aplikasi LP memiliki kendala non negatif. Tetapi, prosedur solusi LP dapat menangani nilai variabel negatif, jika karena beberapa alasan masalah itu menghendaknya dan modelnya dirumuskan dengan tepat. Nilai variabel negatif sering terjadi jika variabel itu merupakan suatu “tingkat” seperti tingkat pertumbuhan dan tingkat inflasi yang dapat naik atau turun. Penurunannya ditunjukkan oleh nilai negatif.

Kendala dituliskan sebagai pertidaksamaan (\leq) dan bukan persamaan ($=$). Persamaan secara tak langsung mengatakan bahwa yang Mungkin timbul pertanyaan mengapa kendala dituliskan sebagai pertidaksamaan seluruh kapasitas produksi digunakan, sementara pertidaksamaan memperbolehkan penggunaan kapasitas secara penuh maupun penggunaan sebagian kapasitas. Dalam beberapa kasus suatu solusi dengan mengizinkan adanya kapasitas

produksi tak terpakai akan memberikan solusi yang lebih baik, yang berarti keuntungan lebih besar, dari pada solusi yang membutuhkan penggunaan semua sumber daya. Jadi, pertidaksamaan menunjukkan adanya keluwesan.

Sekarang masalah LP yang lengkap dapat diringkas sebagai suatu model matematika,

$$\text{Maksimumkan } Z = 3X_1 + 5X_2 + 2X_3$$

dengan syarat:

$$5X_1 + 2X_2 + 4X_3 \leq 240$$

$$4X_1 + 6X_2 + 3X_3 \leq 400$$

$$X_1, X_2, X_3 \geq 0$$

Dengan mencari solusi model ini untuk nilai variabel X_1 , X_2 , dan X_3 , yang optimum penerimaan total Z akan dimaksimumkan.

Contoh 2: Masalah Kombinasi Tanaman

Seorang petani memiliki lahan seluas 2000 hektar, yang menjadi tiga bagian yang terpisah. Petani ingin menanam tiga jenis tanaman: kacang, padi, dan jagung. Ukuran masing-masing lahan, luas areal maksimum untuk setiap jenis tanaman, dan keuntungan per hektar setiap jenis tanaman ditunjukkan seperti berikut:

Tabel 2.2a

Lahan	Luas (ha)
1	500
2	800
3	700
Jumlah 2000	

Tabel 2.2b

Tanaman	Luas Maksimal	Laba/ha
Kacang	900	600
Padi	700	450
Jagung	1000	300

Masing-masing tanaman dapat ditanam pada lahan mana pun, namun, sekurang-kurangnya 60% dari masing-masing lahan harus dibudidayakan. Di samping itu petani menghendaki bahwa proporsi masing-masing lahan yang dibudidayakan adalah sama. Pada khusus ini masalah yang dihadapi petani adalah berapa luas areal setiap jenis tanaman pada setiap lahan agar diperoleh profit maksimum.

a) **Variabel keputusan**

Variabel masalah ini adalah luas masing-masing tanaman yang ditanam pada masing-masing lahan. Luas ini dilambangkan dengan

$$X = \text{luas areal tanaman } i \text{ yang ditanam pada lahan } j$$

di mana

i : 1 (kacang), 2 (padi), 3 (jagung)

j : 1 (lahan 1), 2 (lahan 2), 3 (lahan 3) ini berarti terdapat 9 variabel.

b) **Fungsi tujuan**

Tujuan petani adalah memaksimumkan keuntungan. Total laba merupakan perkalian antara laba per hektar masing-masing jenis tanaman dengan luas masing-masing tanaman yang ditanam pada ketiga lahan. Sehingga fungsi tujuan yang mencerminkan keuntungan yang diperoleh dari setiap tanaman dituliskan sebagai

$$Z = 600 (X_{11} + X_{12} + X_{13}) + 450 (X_{21} + X_{22} + X_{23}) + 300 (X_{31} + X_{32} + X_{33})$$

c) **Sistem kendala**

Kendala pertama adalah batas atas dan batas bawah luas yang dibudidayakan untuk masing-masing lahan. Batas atasnya adalah luas masing-masing lahan itu sendiri, sementara batas bawahnya adalah bahwa 60% masing-masing lahan harus dibudidayakan. Ini menghasilkan kendala-kendala seperti berikut:

$$60\% \times 500 \leq X_{11} + X_{21} + X_{31} \leq 500$$

$$60\% \times 800 \leq X_{12} + X_{22} + X_{32} \leq 800$$

$$60\% \times 700 \leq X_{13} + X_{23} + X_{33} \leq 700$$

Karena kendala-kendala ini tidak sesuai bentuk LP yang biasa (semua variabel pada sisi kiri pertidaksamaan dan jumlah sumber daya di sebelah kanan), mereka harus ditransformasikan. Ini dicapai dengan mengubah masing-masing kendala menjadi dua kendala yaitu luas lahan maksimum dan minimum, hasilnya berupa 6 kendala:

$$X_{11} + X_{21} + X_{31} \geq 300$$

$$X_{11} + X_{21} + X_{31} \leq 500$$

$$X_{12} + X_{22} + X_{32} \geq 480$$

$$X_{12} + X_{22} + X_{32} \leq 800$$

$$X_{13} + X_{23} + X_{33} \geq 420$$

$$X_{13} + X_{23} + X_{33} \leq 420$$

Pembatas lain adalah luas lahan maksimum untuk masing masing tanaman yaitu:

$$X_{11} + X_{12} + X_{13} \leq 900$$

$$X_{21} + X_{22} + X_{23} \leq 700$$

$$X_{31} + X_{32} + X_{33} \leq 1000$$

Kendala terakhir adalah bahwa proporsi masing-masing lahan yang ditanami adalah sama terhadap luas lahan. Kendala ini dihitung rasio seperti berikut:

$$\frac{X_{11} + X_{21} + X_{31}}{500} = \frac{X_{12} + X_{22} + X_{32}}{800} = \frac{X_{13} + X_{23} + X_{33}}{700}$$

Nama persamaan ini tidak dalam bentuk yang pantas dari suatu masalah LP, maka persamaan ini harus ditransformasikan dengan perkalian silang, yang menghasilkan tiga kendala seperti berikut:

$$800(X_{11} + X_{21} + X_{31}) - 500(X_{12} + X_{22} + X_{32}) = 0$$

$$700(X_{12} + X_{22} + X_{32}) - 800(X_{13} + X_{23} + X_{33}) = 0$$

$$700(X_{11} + X_{21} + X_{31}) - 500(X_{13} + X_{23} + X_{33}) = 0$$

Melalui pengamatan yang cermat terhadap tiga kendala ini, tampak bahwa kendala ketiga adalah berlebih. Dalam arti, jika proporsi lahan 1 sama dengan proporsi lahan 2, dan proporsi lahan 2 sama dengan proporsi lahan 3, kemudian proporsi lahan 1 sudah pasti sama dengan proporsi lahan 3 dan persamaan yang mencerminkan rasio ini dapat dihilangkan. Sehingga model LP yang lengkap adalah:

Maks,

$$Z = 600(X_{11} + X_{12} + X_{13}) + 450(X_{21} + X_{22} + X_{23}) + 300(X_{31} + X_{32} + X_{33})$$

Dengan Syarat:

$$X_{11} + X_{21} + X_{31} \geq 300$$

$$X_{11} + X_{21} + X_{31} \leq 500$$

$$X_{12} + X_{22} + X_{32} \geq 480$$

$$X_{12} + X_{22} + X_{32} \leq 800$$

$$X_{13} + X_{23} + X_{33} \leq 420$$

$$X_{13} + X_{23} + X_{33} \leq 700$$

$$X_{11} + X_{12} + X_{13} \leq 900$$

$$X_{21} + X_{22} + X_{23} \leq 700$$

$$X_{31} + X_{32} + X_{33} \leq 100$$

$$800(X_{11} + X_{21} + X_{31}) - 500(X_{12} + X_{22} + X_{32}) = 0$$

$$700(X_{12} + X_{22} + X_{32}) - 800(X_{13} + X_{22} + X_{33}) = 0$$

$$X_{ij} \geq 0 \text{ untuk semua } i \text{ dan } j$$

Contoh 3: Masalah Diet

Untuk menjaga kesehatan, seseorang harus memenuhi kebutuhan minimum perhari akan beberapa zat makanan. Misalnya hanya ada 3 zat makanan yang dibutuhkan, yaitu kalsium, protein dan vitamin A. kebutuhan minimum perhari akan zat-zat makanan tersebut disajikan pada tabel berikut.

Tabel 2.3

Kandungan	Jenis Makanan			Kebutuhan Minimum
	Sayur	Daging	Susu	
Kalsium	5	1	0	8
Protein	2	2	1	10
Vitamin A	1	5	4	22
Harga Per Unit	0,5	0,8	0,6	

Pada kasus ini masalah yang dihadapi orang ini adalah bagaimana mencari kombinasi ketiga jenis makanan itu agar memenuhi kebutuhan minimum perhari dan memberikan biaya terendah.

a) **Variabel keputusan**

Masalah ini terdiri dari 3 variabel yang menunjukkan jumlah masing-masing jenis makanan yang ditempatkan dalam menu, yaitu:

X_1 = jumlah sayur

X_2 = jumlah daging

X_3 = jumlah susu

b) **Fungsi tujuan**

Tujuan masalah ini adalah meminimumkan biaya total menu perhari. Biaya total dalam konteks ini adalah jumlah biaya dari masing-masing jenis makanan yang disajikan dalam menu. Sehingga biaya total, Z , dituliskan sebagai

$$Z = 0,5X_1 + 0,8X_2 + 0,6X_3$$

c) **Sistem kendala**

Dalam masalah ini, kendala adalah kebutuhan minimum akan zat-zat makanan perhari yang telah ditetapkan oleh ahlinya. Kendala untuk kalsium adalah:

$$5X_1 + X_2 \geq 8$$

$5X_1$ adalah sumbangan kalsium dari sayur X_2 adalah sumbangan dari kalsium daging

Pada contoh ini digunakan pertidaksamaan " \geq " yang menunjukkan jumlah minimum kalsium yang dibutuhkan. Dengan kata lain, sekurang-kurangnya 8 unit kalsium yang harus dipenuhi. Pada bentuk LP yang paling umum dan sederhana, masalah maksimisasi menggunakan pertidaksamaan " \leq " sementara dalam minimisasi pertidaksamaan " \geq " yang digunakan. Namun, ini bukan suatu aturan yang mutlak karena masalah-masalah yang lebih kompleks sering memiliki kendala \leq dan \geq maupun persamaan sekaligus.

Kendala untuk protein dan Vitamin A dibentuk dengan cara serupa, yaitu:

$$2X_1 + 2X_2 + X_3 \geq 10 \text{ protein}$$

$$X_1 + 5X_2 + 4X_3 \geq 22 \text{ vitamin A}$$

Masalah LP yang lengkap dapat diringkas menjadi:

$$\text{Minimumkan } Z = 0,5X_1 + 0,8X_2 + 0,6X_3$$

Dengan syarat:

$$\begin{aligned} SX_1 + X_2 &\geq 8 \\ 2X_1 + 2X_2 + X_3 &\geq 10 \\ X_1 + SX_1 + 4X_3 &\geq 22 \\ X_1, X_2, X_3, &\geq 0 \end{aligned}$$

Dengan menyelesaikan model ini untuk variabel keputusan X_1 , X_2 , dan X_3 orang tersebut akan mendapatkan biaya total minimum yang mungkin dengan tetap memenuhi kebutuhan minimum akan kalsium, protein dan Vitamin A. jika solusi ini diikuti, berarti setiap menu orang itu tidak berubah, ini mungkin membosankan.

Contoh 4: Masalah Transportasi

Bulog ingin mengangkut beras dari dua daerah surplus ketiga daerah yang kekurangan. Pasokan dari daerah surplus dan permintaan dari daerah yang kekurangan serta ongkos angkut per unit masing-masing jalur disajikan berikut:

		Daerah Tujuan			Pasokan
		DKI	Kaltim	NTT	
Daerah Asal	Jatim	8	5	6	120
	Sulsel	15	10	12	80
Permintaan		100	70	60	230
					200

Pada kasus ini masalah yang dihadapi bulog adalah menentukan pola penqmnrnan (distribusi) demikiari rupa sehingga biaya transportasi total dapat diminimumkan. Agar masalah ini dapat dirumuskan dalam model LP diperlukan dengan asumsi. Pertama, hanya ada satu jenis beras sehingga kekurangan dapat diisi dari sumber mana saja atau kelebihan dapat dikirim ke mana saja. Kedua, ongkos angkut per unit adalah tetap atau tidak dipengaruhi oleh jumlah unit yang diangkut.

a) Variabel keputusan

Apa yang harus ditemukan di sini adalah volume beras yang diangkut dari setiap daerah surplus menuju setiap daerah kekurangan. Misalkan:

X_{ij} adalah volume beras dari sumber i ke tujuan j

Di mana

$i = 1$ (Jatim), 2 (sulsel)

$j = 1$ (DKI), 2 (Kaltim), 3 (NTT), berarti terdapat 6 variabel

b) **Fungsi tujuan**

Tujuan bulog adalah menekankan biaya transpor total, yaitu:

$$Z = 8X_{11} + 5X_{12} + 6X_{13} + 15X_{21} + 10X_{22} + 12X_{23}$$

c) **Sistem kendala**

Di sini jumlah permintaan lebih besar dari pada pasokan, ini berarti seluruh pasokan akan dihabiskan, sehingga ada kendala pasokan bertanda $=$. Sementara itu tidak semua permintaan dapat dipenuhi, sehingga kendala permintaan bertanda \leq model LP masalah itu menjadi:

minimumkan

$$Z = 8X_{11} + 5X_{12} + 6X_{13} + 15X_{21} + 10X_{22} + 12X_{23}$$

Dengan syarat:

$$\left. \begin{array}{l} X_{11} + X_{12} + X_{13} = 20 \\ X_{21} + X_{22} + X_{23} = 80 \end{array} \right\} \text{Kendala pasokan}$$

$$\left. \begin{array}{l} X_{11} + X_{21} \leq 100 \\ X_{12} + X_{22} \leq 70 \\ X_{13} + X_{23} \leq 60 \end{array} \right\} \text{Kendala permintaan}$$

Pengamatan teliti pada kendala mengatakan bahwa semua koefisien sama dengan satu. Ini bukan kebetulah, tetapi merupakan ciri khas masalah transportasi.

2.2.6 Asumsi Model LP

Model LP mengandung asumsi-asumsi implisit tertentu yang harus dipenuhi agar definisinya sebagai suatu masalah LP menjadi absah. Asumsi itu menuntut bahwa hubungan fungsional dalam masalah itu adalah linier dan adaptif, dapat dibagi dan deterministik. Berikut ini akan diterangkan lebih rinci.

a) *Linierity* dan *Additivity*

Syarat utama dari LP adalah bahwa fungsi tujuan dan semua kendala harus linier. Dengan kata lain, jika suatu kendala melibatkan dua variabel keputusan, dalam diagram dimensi dua ia akan berupa suatu garis lurus. Begitu juga, suatu kendala yang melibatkan tiga variabel akan menghasilkan suatu bidang datar dan kendala yang melibatkan n variabel akan menghasilkan *hyperplane* (bentuk geometris yang rata) dalam ruang berdimensi n .

Kata linier secara tidak langsung mengatakan bahwa hubungannya proporsional yang berarti bahwa tingkat perubahan atau kemiringan hubungan fungsional itu adalah konstan dan karena itu perubahan nilai variabel akan mengakibatkan perubahan relatif nilai fungsi dalam jumlah yang sama.

LP juga mensyaratkan bahwa jumlah variabel kriteria dan jumlah penggunaan sumber daya harus bersifat aditif. Contohnya, keuntungan total Z , yang merupakan variabel kriteria, sama dengan jumlah keuntungan yang diperoleh dari masing-masing kegiatan, $C_i X_j$. Juga, seluruh sumber daya yang digunakan untuk semua kegiatan, harus sama dengan jumlah sumber daya yang digunakan untuk masing-masing kegiatan.

Aditif dapat diartikan sebagai tak adanya penyesuaian pada perhitungan variabel kriteria karena terjadinya interaksi. Contohnya, dalam masalah kombinasi produk disebutkan bahwa keuntungan per unit produk 1 adalah Rp3,- produk 2 sebesar Rp5,- dan produk 3 sebesar Rp2,- jika masing-masing produk dijual secara terpisah. Bisa jadi, jika ketiga produk dijual secara serentak pada daerah yang sama dapat menyebabkan penurunan keuntungan, sehingga perlu memasukan penyesuaian interaksi ke dalam variabel kriteria, misalnya saja menjadi:

$$Z = 3X_1 + 5X_2 + 2X_3 - X_1 X_2 X_3$$

Model terakhir ini mengandung suku yang bersifat multiplikatif sehingga tidak linier, dan metode LP tak dapat menangani masalah demikian.

b) *Divisibility*

Asumsi ini berarti bahwa nilai solusi yang diperoleh, X_j , tidak harus berupa bilangan bulat. Ini berarti nilai X_j , dapat terjadi pada nilai pecah manapun. Karena itu variabel keputusan merupakan variabel kontinu, sebagai lawan dari variabel diskrit atau bilangan bulat.

Pada contoh masalah kombinasi produk, akan tidak masuk akal jika harus memproduksi produk 1 (katakan kapal), misalnya saja sebanyak 2,75. Akibatnya, jika nilai-nilai bulat mutlak diperlukan, suatu model LP alternatif, yaitu *Integer Programming* harus digunakan.

c) *Deterministic*

Dalam LP, semua parameter model (c_j , a_{ij} , dan b_i) diasumsikan diketahui konstan. LP secara tak langsung mengasumsikan suatu masalah keputusan dalam suatu kerangka statis di mana semua parameter diketahui dengan kepastian. Dalam Kenyataannya, parameter model jarang bersifat deterministik, karena mereka mencerminkan kondisi masa depan maupun sekarang, dan keadaan masa depan jarang diketahui dengan pasti.

Ada beberapa cara untuk mengatasi ketidakpastian parameter dalam model LP. Analisis sensitivitas adalah suatu teknik yang dikembangkan untuk menguji nilai solusi, bagaimana kepekaannya terhadap perubahan-perubahan parameter.

2.3 Solusi Pada Program Linear

2.3.1 Penyelesaian Grafik Model LP

Masalah LP dapat diilustrasikan dan dipecahkan secara grafik jika ia hanya memiliki dua variabel keputusan. Meski masalah-masalah dengan dua variabel jarang terjadi dalam dunia nyata, penafsiran geometris dari metode grafis ini sangat bermanfaat. Dari sini, kita dapat menarik kesimpulan yang akan menjadi dasar untuk pembentukan metode pemecahan (solusi) yang umum melalui algoritma simpleks.

Pikirkan masalah kombinasi produksi yang sederhana seperti berikut. Suatu, perusahaan menghasilkan dua barang, meja dan kursi. Harga masing-masing barang dan kebutuhan sumber daya terlihat pada tabel berikut. Di samping itu, menurut bagian penjualan, permintaan meja tidak akan melebihi 4 unit.

Sumber Daya	Meja	Kursi	Sumber Daya yang Tersisa
Bahan Mentah	1	2	10
Buruh	6	6	36
Harga Per Unit	4	5	

Masalah untuk memaksimalkan penerimaan dirumuskan menjadi:

Maksimalkan

$$Z = 4X_1 + 5X_2$$

dengan syarat:

$$X_1 + 2X_2 \leq 10$$

$$6X_1 + 6X_2 \leq 36$$

$$X_1 \leq 4$$

$$X_1, X_2 \geq 0$$

Model itu disajikan dengan grafik pada Gambar 2.1. Untuk menggambarkan ketiga kendala pertidaksamaan, perlu memperlakukan masing-masing sebagai suatu persamaan.

Suatu cara sederhana untuk menggambarkan masing-masing persamaan garis adalah dengan menetapkan salah satu variabel dalam suatu persamaan sama dengan.

2.4 Masalah Teknis dalam Program Linear Metode Grafik

2.4.1 Kasus Khusus Model LP a

a) **Solusi Optimum Berganda**

Solusi optimum berganda akan terjadi dalam suatu masalah LP jika fungsi tujuan terletak pada lebih dari satu titik optimum. Situasi ini terjadi jika kemiringan fungsi tujuan dan salah satu persamaan kendala adalah sama.

Maksimalkan

$$Z = 4X_1 + 4X_2$$

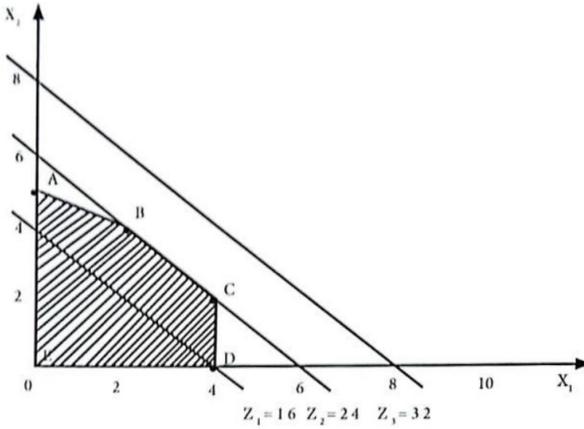
Dengan syarat:

$$X_1 + 2X_2 \leq 10$$

$$6X_1 + 6X_2 \leq 36$$

$$X_1 \leq 4$$

$$X_1, X_2 \geq 0$$



Gambar 2.1

b) **Masalah Tak Layak**

Dalam beberapa kasus suatu masalah LP mungkin tidak memiliki solusi yang layak. Dengan kata lain, tak ada titik-titik yang secara serentak memenuhi semua kendala dalam masalah itu.

Maksimumkan

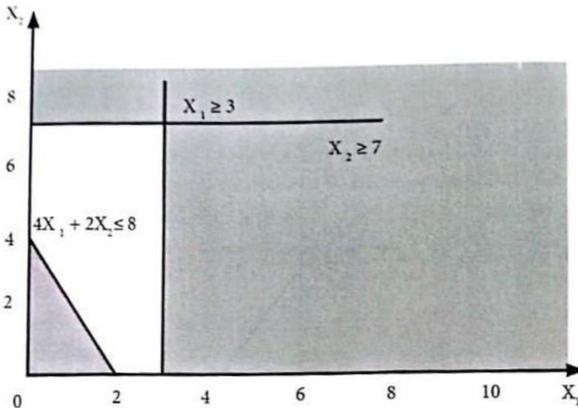
$$Z = 5X_1 + 3X_2$$

Dengan Syarat:

$$4X_1 + 2X_2 \leq 8 \quad X_1 \geq 3$$

$$X_2 \geq 7$$

$$X_1, X_2 \geq 0$$



Gambar 2.2

c) **Masalah Tak Terbatas**

Dalam beberapa masalah ruang solusi yang layak dibentuk oleh kendala-kendala yang tidak dibatasi dalam suatu batas yang tertutup. Dalam hal ini, fungsi tujuan dapat meningkat tak terbatas tanpa pernah mencapai batas maksimumnya karena ia tak pernah mencapai batas kendala.

Maksimumkan

$$Z = 4X_1 + 2X_2$$

Dengan Syarat:

$$-X_1 + 2X_2 \leq 6$$

$$-X_1 + X_2 \leq 6$$

$$X_1, X_2 \geq 0$$

2.4.2 Metode Grafik

Metode grafik hanya dapat digunakan dalam pemecahan masalah LP yang ber"dimensi" $2 \times n$ atau $m \times 2$, karena keterbatasan kemampuan suatu grafik dalam menyampaikan sesuatu.

Langkah-langkah dalam menggunakan metode grafik:

1. Menentukan fungsi tujuan dan memformulasikannya dalam bentuk matematis.
2. Mengidentifikasi batasan-batasan yang berlaku dan memformulasikannya dalam bentuk matematis.
3. Menggambar masing-masing garis fungsi batasan dalam satu sistem salib sumbu.
4. Mencari titik yang paling menguntungkan (optimal) dihubungkan dengan fungsi tujuan.

2.5 Soal dan Kasus

2.5.1 Contoh Soal Metode Grafik

Perusahaan makanan ROYAL merencanakan untuk membuat dua jenis makanan yaitu Royal Bee dan Royal Jelly. Kedua jenis makanan tersebut mengandung vitamin dan protein. Royal Bee paling sedikit diproduksi 2 unit dan Royal Jelly paling sedikit diproduksi 1 unit. Tabel berikut menunjukkan jumlah vitamin dan protein dalam setiap jenis makanan:

Jenis Makanan	Vitamin(Unit)	Protein(Unit)	Biaya Per Unit(Rp)
Royal Bee	2	2	100
Royal jelly	1	3	80
Minuman Kebutuhan	8	12	-

Bagaimana menentukan kombinasi kedua jenis makanan agar meminimumkan biaya produksi.

Langkah-langkah:

- Menentukan variabel

$X_1 = \text{royal bee}; X_2 = \text{royal jelly}$

- 1] Fungsi Tujuan

$Z_{min} = 100X_1 + 80X_2$

- 2] Fungsi Kendala

$2X_1 + X_2 \dots\dots\dots 8$ Vitamin

$2X_1 + 3X_2 \dots\dots\dots 12$ Protein

$X_1 \dots\dots\dots 2$

$X_2 \dots\dots\dots 1$

- Membuat Grafik

a| $2X_1 + X_2 = 8$

$X_1 = 0, X_2 = 8$

$X_2 = 0, X_1 = 4$

$$b) \quad 2X_1 + 3X_2 = 12$$

$$X_1 = 0, X_2 = 4$$

$$X_2 = 0, X_1 = 6$$

$$c) \quad X_1 = 2$$

$$d) \quad X_2 = 1$$

- Solusi optimal tercapai pada titik B (terdekat dengan titik origin), yaitu persilangan garis kendala (1) dan (2)

$$2X_1 + X_2 = 8$$

$$2X_1 + 3X_2 = 12$$

$$-2X_2 = -4$$

$$X_2 = 2$$

- Masukkan X_2 ke kendala (1) $2X_1 + X_2 = 8$

$$2X_1 + 2 = 8$$

$$2X_1 = 8 - 2$$

$$2X_1 = 6$$

$$X_1 = 3$$

- Masukkan nilai X_1 dan X_2 ke Z

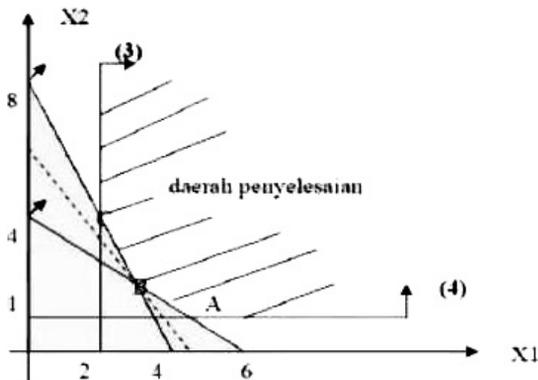
$$Z_{min} = 100(3) + 80(2)$$

$$Z_{min} = 300 + 160$$

$$Z_{min} = 460$$

- Kesimpulan:

Untuk meminimumkan biaya produksi, maka $X_1 = 3$ dan $X_2 = 2$ dengan biaya produksi Rp460.000,00.



2.5.2 Contoh Kasus

Seorang pengrajin menghasilkan satu tipe meja dan satu tipe kursi. Proses yang dikerjakan hanya merakit meja dan kursi. Dibutuhkan waktu 2 jam untuk merakit 1 unit meja dan 30 menit untuk merakit 1 unit kursi. Perakitan dilakukan oleh 4 orang karyawan dengan waktu kerja 8 jam perhari. Pelanggan pada umumnya membeli paling banyak 4 kursi untuk 1 meja. Oleh karena itu pengrajin harus memproduksi kursi paling banyak empat kali jumlah meja. Harga jual per unit meja adalah Rp1,2 juta dan per unit kursi adalah Rp500ribu. Formulasikan kasus tersebut ke dalam model matematiknya!

Penyelesaian

- Langkah Pertama

Mengidentifikasi tujuan, alternatif keputusan dan sumber daya yang membatasi. Berdasarkan informasi yang diberikan pada soal, tujuan yang ingin dicapai adalah memaksimalkan pendapatan. Alternatif keputusan adalah jumlah meja dan kursi yang akan diproduksi. Sumber daya yang membatasi adalah waktu kerja karyawan dan perbandingan jumlah kursi dan meja yang harus diproduksi (pangsa pasar).

- Langkah Kedua

Memeriksa sifat *proporsionalitas*, *additivitas*, *divisibilitas* dan kepastian. Informasi di atas tidak menunjukkan adanya pemberian diskon, sehingga harga jual per meja maupun kursi akan sama meskipun jumlah yang dibeli semakin banyak. Hal ini mengisyaratkan bahwa total pendapatan yang diperoleh pengrajin proporsional terhadap jumlah produk yang terjual. Penggunaan sumber daya yang membatasi, dalam hal ini waktu kerja karyawan dan pangsa pasar juga proporsional terhadap jumlah meja dan kursi yang diproduksi. Dengan demikian dapat dinyatakan sifat proporsionalitas dipenuhi. Total pendapatan pengrajin merupakan jumlah pendapatan dari keseluruhan meja dan kursi yang terjual. Penggunaan sumber daya (waktu kerja karyawan dan pangsa pasar) merupakan penjumlahan waktu yang digunakan untuk memproduksi meja dan kursi. Maka dapat dinyatakan juga sifat additivitas dipenuhi. Sifat divisibilitas dan kepastian juga dipenuhi.

- Langkah Ketiga

Ada dua variabel keputusan dan dua sumber daya yang membatasi. Fungsi tujuan merupakan maksimisasi, karena semakin besar pendapatan akan semakin disukai oleh pengrajin. Fungsi kendala pertama (batasan waktu) menggunakan pertidaksamaan \leq , karena waktu yang tersedia dapat digunakan sepenuhnya atau tidak, tapi tidak mungkin melebihi waktu yang ada. Fungsi kendala yang kedua bisa menggunakan \leq atau \geq tergantung dari pendefinisian variabelnya.

Definisikan Variabelnya:

X_1 = jumlah meja yang akan diproduksi

X_2 = jumlah kursi yang akan diproduksi

Model umum Pemrograman Linier kasus di atas adalah:

Fungsi tujuan:

Maksimumkan $Z = 1.2X_1 + 0.5X_2$

Kendala: $2X_1 + 0.5X_2 \leq 32$

$X_1/X_2 \geq 1/4$ atau $4X_1 \geq X_2$ atau $4X_1 - X_2 \geq 0$

$X_1, X_2 \geq 0$

Bab 3

Program Linear Metode Simpleks

3.1 Metode Simpleks

3.1.1 Pengertian Metode Simpleks

Menurut Eddy Herjanto (2008:51) Metode simpleks adalah suatu metode yang secara sistematis dimulai dari suatu penyelesaian dasar yang fisibel ke penyelesaian dasar fisibel lainnya, yang dilakukan berulang-ulang (*iterative*) sehingga tercapai suatu penyelesaian optimum (Setiawan Budi Tri Wahyono, Iskandar, 2017).

Metode simpleks merupakan salah satu teknik penyelesaian dalam program linier yang digunakan sebagai teknik pengambilan keputusan dalam permasalahan yang berhubungan dengan pengalokasian sumber daya secara optimal. Metode simpleks digunakan untuk mencari nilai optimal dari program linier yang melibatkan banyak *constraint* (pembatas) dan banyak variabel (lebih dari dua variabel). Penemuan metode ini merupakan lompatan besar dalam riset operasi dan digunakan sebagai prosedur penyelesaian dari setiap program komputer.

Program Linier merupakan metode matematik dalam mengalokasikan sumber daya yang langka untuk mencapai tujuan tunggal seperti memaksimalkan keuntungan atau meminimumkan biaya. LP (Linear Programming) banyak diterapkan dalam membantu menyelesaikan masalah ekonomi, industri, militer, sosial dan lain-lain.

Dari berbagai metode penyelesaian program linier, metode simpleks merupakan metode yang paling ampuh dan terkenal. Metode simpleks didasarkan atas pengertian bahwa solusi optimal dari masalah program linier, jika ada, selalu dapat ditemukan di salah satu dari “solusi dasar yang berlaku”. Oleh sebab itu dalam metode simpleks, langkah pertama adalah untuk memperoleh solusi dasar yang berlaku.

3.1.2 Penjelasan Metode Simpleks

Metode simpleks merupakan sebuah cara untuk menyelesaikan soal pemrograman linear dimana pengulangan prosedur matematis itu dilakukan untuk menguji titik-titik sudut sehingga ditemukan penyelesaian optimal. Simpleks adalah sebuah prosedur matematis untuk menemukan penyelesaian optimal soal pemrograman linear dengan cara menguji titik-titik sudutnya (Siswanto, 2007). “Metode Simpleks adalah suatu metode yang secara sistematis dimulai dari suatu penyelesaian dasar yang visibel ke penyelesaian dasar fisibel lainnya, secara berulang-ulang sehingga tercapai suatu penyelesaian dasar yang optimum” (Rosita, 2019; Siringoringo, 2005).

Beberapa istilah yang digunakan dalam metode simpleks, penjelasannya diantaranya sebagai berikut:

a) Iterasi

Tahapan perhitungan dimana nilai dalam perhitungan itu tergantung dari nilai tabel sebelumnya.

b) Variabel Non Basis

Variabel yang nilainya diatur menjadi nol pada sembarang iterasi. Dalam terminology umum, jumlah variabel non basis selalu sama dengan derajat bebas dalam sistem persamaan.

c) Variabel Basis

Variabel yang nilainya bukan nol pada sembarang iterasi. Pada solusi awal, variabel basis merupakan variabel *slack* (jika fungsi kendala menggunakan pertidaksamaan $<$) atau variabel buatan (jika fungsi kendala menggunakan pertidaksamaan $>$ atau $=$). Secara umum, jumlah variabel batas selalu sama dengan jumlah fungsi pembatas (tanpa fungsi non negatif).

d) Solusi atau Nilai Kanan (NK)

Nilai sumber daya pembatas yang masih tersedia. Pada solusi awal, nilai kanan atau solusi sama dengan jumlah sumber daya pembatas awal yang ada, karena aktivitas belum dilaksanakan.

e) Variabel Slack

Variabel yang ditambahkan ke model matematik kendala untuk mengkonversikan pertidaksamaan $<$ menjadi persamaan ($=$). Penambahan variabel ini terjadi pada tahap inisialisasi. Pada solusi awal, variabel *slack* akan berfungsi sebagai variabel basis.

f) Variabel Surplus

Variabel yang dikurangkan dari model matematik kendala untuk mengkonversi pertidaksamaan $>$ menjadi persamaan ($=$). Penambahan variabel ini terjadi pada tahap inisialisasi. Pada solusi awal, variabel surplus tidak dapat berfungsi sebagai variabel bebas.

g) Variabel Buatan

Variabel yang ditambahkan ke model matematik kendala dengan bentuk $>$ atau $=$ untuk difungsikan sebagai variabel basis awal. Penambahan variabel ini terjadi pada tahap inisialisasi. Variabel ini harus bernilai 0 pada solusi optimal, karena kenyataannya variabel ini tidak ada. Variabel ini hanya ada di atas kertas.

h) Kolom Pivot (Kolom Kerja)

Kolom yang memuat variabel masuk. Koefisien pada kolom ini akan menjadi pembagi nilai kanan untuk menentukan baris pivot (baris kerja).

i) Baris Pivot

Salah satu baris dari antara variabel baris yang memuat variabel keluar.

j) Elemen Pivot

Elemen yang terletak pada perpotongan kolom dan baris pivot. Elemen pivot akan menjadi dasar perhitungan untuk tabel Simpleks berikutnya.

k) Variabel Masuk

Variabel yang terpilih untuk menjadi variabel baris pada iterasi berikutnya. Variabel masuk dipilih satu dari antara variabel non basis pada setiap iterasi. Variabel ini pada iterasi berikutnya akan bernilai positif.

l) Variabel Keluar

Variabel yang keluar dari variabel basis pada iterasi berikutnya dan digantikan dengan variabel masuk. Variabel keluar dipilih satu dari antara variabel basis pada setiap iterasi dan bernilai nol.

**Tabel 3.1 Bentuk Umum Tabel Analisis Program Maksimisasi
(Haming and A1, 2017)**

C_1	Mix		a_1	a_2	$a \dots$	a_n	0	0	0
		Q	X_1	X_2	$X \dots$	X_n	S_1	S_2	S_n
0	S_1	b_1	c_{11}	c_{12}	$c \dots$	c_{mn}	1	0	0
0	S_2	b_2	c_{21}	c_{22}	$c \dots$	c_{mn}	0	1	0
0	S_3	b_3	c_{31}	c_{32}	$c \dots$	c_{mn}	0	0	1
Z_j									
$Z_j - C_j$									

Legenda:

Fungsi tujuan adalah:

Maksimumkan

$$Z = a_1X_1 + a_2X_2 + \dots + a_nX_n + 0S_1 + 0S_2 + \dots + 0S_n$$

Dengan kendala:

$$C_{11}X_1 + C_{12}X_2 + \dots + C_{mn}X_n + 1S_1 + 0S_2 + \dots + 0S_n = b_1$$

$$C_{21}X_1 + C_{22}X_2 + \dots + C_{mn}X_n + 0S_1 + 1S_2 + \dots + 0S_n = b_2$$

$$C_{31}X_1 + C_{32}X_2 + \dots + C_{mn}X_n + 0S_1 + 0S_2 + \dots + 1S_n = b_3$$

C_j = Kontribusi unit dari fungsi tujuan, yaitu a untuk variabel keputusan dan 0 untuk variabel slack S_j

C_{ij} = Koefisien fungsi kendala

Z_j = Kontribusi pada setiap proses iterasi

Tabel terdiri atas bagian-bagian penting sebagai berikut:

1. Kepala tabel, dibagi atas dua bagian. Bagian atas kepala tabel dipakai sebagai tempat menuliskan kontribusi unit (C_j) fungsi tujuan (karena itu disebut juga *objective row*), sedangkan bagian bawah kepala tabel dipakai sebagai tempat menuliskan semua nama peubah keputusan X_i dan perubahan dummy S_j (karena itu disebut juga *variable row*). Kepala tabel ini dibagi atas beberapa kolom, yaitu: C_j (untuk kontribusi unit fase iterasi). Kolom *product mix* (sesuai bauran setiap proses iterasi), kolom Q (nilai sisi kanan, b_j fungsi kendala), kolom peubah keputusan (d disesuaikan dengan jumlah peubah menurut fungsi tujuan), serta kolom peubah dummy S_j (d disesuaikan dengan jumlah baris kendala).

2. Badan tabel disebut juga *problem rows*, yaitu tempat menuliskan koefisien fungsi kendala dan koefisien peubah dummy S_j selanjutnya, juga tempat mencatat hasil proses iterasi mulai tahapan pertama sampai tahap optimal.
3. Kaki tabel, terbagi atas dua baris, yaitu baris Z_j dan baris $Z_j - C_j$. Baris Z_j disebut juga *index row* adalah baris tempat mencatat hasil perkalian vector C_j dengan vektor kolom yang ada dalam badan tabel. Baris $Z_j - C_j$ adalah baris tempat mencatat hasil pengurangan baris Z_j dengan koefisien fungsi tujuan yang ada pada bagian atas kepala tabel dan baris ini juga disebut *identity row* (baris identitas). Disebut demikian karena baris ini menjadi landasan untuk menentukan berikut ini:
 - a) Kolom kunci (*key column*).
 - b) Tahap optimal analisis (dalam maksimisasi, jika semua tanda dari angka hasil kurang $Z_j - C_j$ sudah positif seluruhnya).

3.1.3 Langkah-langkah Metode Simpleks

Adapun langkah-langkah penyelesaian dengan metode simpleks adalah sebagai berikut menurut (Haslan and Dkk, 2018):

1. Ubah fungsi tujuan dan kendala ke dalam bentuk standar.
2. Susun semua nilai kedalam tabel simpleks.
3. Tentukan kolom kunci (variabel keputusan) yang masuk sebagai variabel basis (*entering variable*). Kolom kunci adalah kolom yang mempunyai nilai pada baris Z (fungsi tujuan) yang bernilai negatif (-) dengan angka terbesar.
4. Tentukan baris kunci, untuk melakukan variabel yang akan keluar dari baris kunci (*leaving variable*). Baris kunci adalah baris dengan nilai indeks positif terkecil, dengan perhitungan indeks sebagai berikut:

$$\text{Indeks} = \frac{\text{Nilai Kanan (KN)}}{\text{Nilai setiap baris pada kolom kunci}}$$

5. Mengubah nilai-nilai pada baris kunci, dengan cara membaginya dengan angka kunci. Angka kunci merupakan nilai yang posisinya berada pada perpotongan antara kolom kunci dengan baris kunci.

$$\text{Nilai baris kunci baru} = \frac{\text{Nilai pada baris kunci lama}}{\text{Angka kunci}}$$

6. Membuat baris baru dengan mengubah nilai-nilai baris (selain baris kunci) sehingga nilai-nilai kolom kunci = 0, dengan mengikuti perhitungan sebagai berikut:

$$\text{Nilai baris baru} = \text{Nilai baris lama} - (KAK \times NBBK)$$

Di mana:

KAK = Koefisien Angka Kolom Kunci (nilai setiap baris kolom kunci)

$NBBK$ = Nilai Baris Kunci

7. Ulangi langkah di atas (langkah 3–6 atau disebut iterasi), sampai tidak terdapat nilai negatif pada baris Z (baris fungsi tujuan).

3.1.4 Contoh Metode Simpleks

Adapun contoh sederhana dari metode simpleks yaitu sebagai berikut:

1. Misalkan Anda memiliki masalah optimasi di mana Anda ingin memaksimalkan keuntungan dari penjualan dua jenis produk, A dan B. Anda memiliki kendala produksi dan kendala bahan baku yang harus dipenuhi. Dalam hal ini, metode Simpleks dapat digunakan untuk menemukan berapa banyak produk A dan B yang harus diproduksi untuk mencapai keuntungan maksimal sambil memenuhi kendala produksi dan bahan baku.
2. Perencanaan Keuangan: Saat mengalokasikan anggaran bulanan, Anda dapat menggunakan metode simpleks untuk menentukan alokasi optimal untuk berbagai kategori pengeluaran, seperti makanan, transportasi, dan rekreasi, dengan mempertimbangkan sumber daya terbatas (pendapatan bulanan).
3. Penjadwalan Waktu: Bagi seorang mahasiswa atau pekerja, mengatur jadwal harian dengan banyak tugas dan keterbatasan waktu dapat dianggap sebagai masalah pemrograman linier. Metode simpleks dapat membantu Anda menentukan cara terbaik untuk mengalokasikan waktu Anda pada berbagai tugas untuk mencapai hasil optimal.

3.2 Bentuk Baku Model Linear Programming

Dalam menggunakan metode simpleks untuk menyelesaikan masalah-masalah Linear Programming, model Linear Programming harus diubah ke dalam suatu bentuk umum yang dinamakan “bentuk baku” (*standard form*). Ciri-ciri bentuk baku model Linear Programming adalah:

- Semua kendala berupa persamaan dengan sisi kanan non negatif.
- Semua variabel non negatif.
- Fungsi tujuan dapat maksimum maupun minimum.

Untuk memudahkan melakukan transformasi ke bentuk baku ikuti contoh berikut:

3.2.1 Kendala

- Suatu kendala jenis \leq (\geq) dapat diubah menjadi suatu persamaan dengan menambahkan suatu variabel *slack* ke (mengurangkan suatu variabel surplus dari sisi kiri kendala). Contoh:

Pada kendala $X_1 + X_2 \leq 15$ ditambahkan suatu slack $S_1 \geq 0$ pada sisi kiri untuk mendapatkan persamaan $X_1 + X_2 + S_1 = 15$. Jika kendala menunjukkan keterbatasan penggunaan suatu sumber daya, S_1 akan menunjukkan *slack* atau jumlah sumber daya yang tak digunakan.

- Pada kendala $3X_1 + 2X_2 - 3X_3 \geq 5$ dikurangkan suatu variabel surplus $S_2 \geq 0$ pada sisi kiri untuk memperoleh persamaan $3X_1 + 2X_2 - 3X_3 - S_2 = 5$
- Sisi kanan suatu persamaan dapat selalu dibuat non negatif dengan cara mengalikan kedua sisi dengan -1 .

Contoh:

$$-5X_1 + X_2 = -25 \text{ adalah ekuivalen secara matematik dengan } 5X_1 - X_2 = 25$$

Arah pertidaksamaan di balik jika kedua sisi dikalikan -1 .

Contoh:

$$-5X_1 + X_2 \leq -25 \text{ dapat diganti dengan } 5X_1 - X_2 \geq 25$$

a) Variabel

Sebagian atau semua variabel dikatakan *unrestricted* jika mereka dapat memiliki nilai negatif maupun positif. Variabel *unrestricted* dapat diekspresikan dalam dua variabel non negatif dengan menggunakan substitusi

$$X_j = X'_j - X''_j$$

di mana

X_j = variabel *unrestricted* dan $X'_j, X''_j \geq 0$

Substitusi ini memengaruhi seluruh kendala dan fungsi tujuan yang akan lebih dijelaskan kemudian.

b) Fungsi Tujuan

Meskipun model Linear Programming dapat berjenis maksimisasi maupun minimisasi, terkadang bermanfaat untuk mengubah salah satu bentuk ke bentuk lain. Maksimisasi dari suatu fungsi adalah ekuivalen dengan minimisasi dari negatif fungsi yang sama, dan sebaliknya.

Contoh:

$$\text{Maks. } Z = 50X_1 + 80X_2 + 60X_3$$

Ekuivalen secara matematik dengan

$$\text{Min } (-Z) = -50X_1 - 80X_2 - 60X_3$$

Ekuivalen berarti bahwa untuk seperangkat kendala yang sama, nilai optimum $X_1, X_2,$ dan $X_3,$ dan adalah sama pada kedua kasus. Perbedaannya hanya pada nilai fungsi tujuan, meski besar angka sama, tetapi tandanya berlawanan.

3.3 Solusi pada Program Linear Metode Simpleks

Terdapat beberapa solusi pada program linear metode simpleks yaitu sebagai berikut:

3.3.1 Solusi Optimum Berganda

Jika fungsi tujuan sejajar dengan suatu kendala, maka akan terjadi nilai optimal yang sama pada lebih dari satu titik solusi. Keadaan ini dinamakan optimum berganda atau optimum alternatif.

Contoh:

Maksimumkan

$$Z = 2X_1 + 4X_2$$

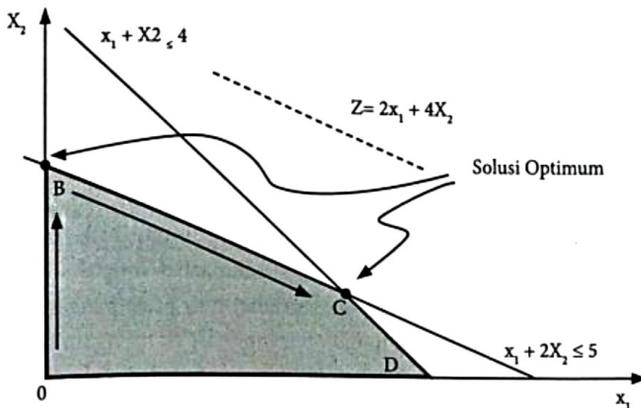
Dengan syarat:

$$X_1 + 2X_2 \leq 5$$

$$X_1 + X_2 \leq 4$$

$$X_1, X_2 \leq 0$$

Penyelesaian grafik masalah ini ditunjukkan pada Gambar 3.2



Gambar 3.1

Tabel simpleks untuk masalah itu adalah:

Tabel 3.2

Basis	X_1	X_2	S_1	S_2	Solusi
Z	-2	-4	0	0	0
S_1	1	2	1	0	5
S_2	1	1	0	1	5

Tabel 3.3 (Optimum)

Basis	X_1	X_2	S_1	S_2	Solusi
Z	0	0	2	0	10
S_1	1/2	1	1/2	0	5/2
S_2	1/2	0	-1/2	1	5/2

Tabel 3.4 (Optimum alternatif)

Basis	X_1	X_2	S_1	S_2	Solusi
Z	0	0	2	0	10
S_1	0	1	1	-1	1
S_2	1	0	-1	2	3

Terjadinya optimum berganda dapat dilihat pada tabel simpleks optimal. Jika koefisien variabel non basis pada persamaan Z (Tabel 3.3) ada yang sama dengan nol, berarti bahwa variabel tersebut dapat masuk menjadi variabel basis tanpa mengubah nilai Z. Pada Tabel 3.2, X_1 menjadi variabel basis menggantikan S_2 . Tabel itu menghasilkan solusi baru yaitu $X_1 = 3$, $X_2 = 1$, $Z = 10$ pada titik C. sementara solusi sebelumnya adalah $X_1 = 0$, $X_2 = 5/2$ dan $Z = 10$ pada titik B.

Metode simpleks bergerak dari satu titik pojok ke titik pojok lain yang berdekatan. Karena itu sesungguhnya titik titik sepanjang garis BC juga memberikan nilai fungsi tujuan yang sama. Dalam praktik, pengetahuan tentang optimum alternative sangat bermanfaat karena ia memberikan kesempatan untuk memilih solusi yang paling cocok dengan situasi yang dihadapi pengambil keputusan.

Berlawanan dengan optimum alternative adalah optimum unik. Ini terjadi jika koefisien variabel non basis pada persamaan Z pada tabel simpleks optimal tidak sama dengan nol. Artinya, setiap perubahan variabel non basis pada tabel optimum akan diikuti perubahan dalam fungsi tujuan. Sehingga tidak mungkin ditemui solusi layak lain yang nilainya sama dengan solusi optimal. Dengan kata lain, optimum unik berarti hanya terdapat satu titik optimum.

3.3.2 Solusi Tak Terbatas

Pada beberapa Linear Programming, nilai variabel mungkin bertambah tak terbatas tanpa menyimpang dari kendala, berarti bahwa ruang solusi menjadi tak terbatas sekurang-kurangnya pada satu arah. Akibatnya, nilai fungsi tujuan dapat bertambah tanpa pernah mencapai batas kendala. Dalam keadaan ini dikatakan bahwa baik ruang solusi maupun nilai optimum adalah tak terbatas.

Sebab ketidakterbatasan itu adalah karena kesalahan membuat model. Kesalahan itu dapat berupa:

1. Satu atau lebih kendala yang tak berlebih tak diikuti sertakan.
2. Parameter dari beberapa kendala tak diduga dengan benar.

Contoh berikut menunjukkan bagaimana cara mengetahui adanya ketidakterbatasan baik dalam ruang solusi maupun nilai fungsi tujuan melalui tabel simpleks.

Maksimumkan

$$Z = 2X_1 + X_2$$

Dengan syarat:

$$X_1 - X_2 \leq 10$$

$$2X_1 \leq 40$$

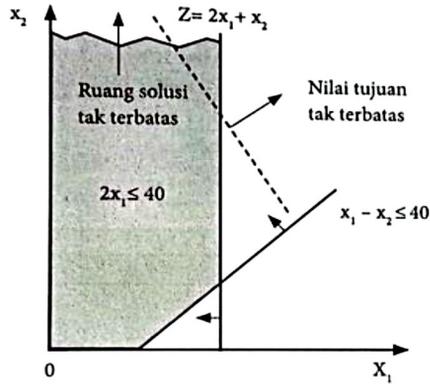
$$X_1, X_2 \geq 0$$

Tabel 3.5

Basis	X_1	X_2	S_1	S_2	Solusi	Rasio
Z	-2	-1	0	0	0	
S_1	1	-1	1	0	10	
S_2	2	0	0	1	40	

Pada Tabel 3.5 baik X_1 dan X_2 adalah calon *entering variable*. Karena X_1 memiliki koefisien negatif terbesar, maka ia yang dipilih. Perhatikan bahwa semua elemen di bawah kolom X_2 adalah non positif, berarti X_2 dapat naik tanpa batas dan tetap memenuhi semua kendala. Karena setiap unit kenaikan X_2 akan meningkatkan fungsi tujuan satu unit, satu kenaikan X_2 yang tak terbatas juga akan meningkatkan Z secara tak terbatas. Sehingga tanpa perhitungan lebih lanjut dapat disimpulkan

bahwa masalah ini tidak memiliki batas solusi. Hasil ini dapat dilihat pada Gambar 3.2. Ruang solusi tak dibatasi ke arah x_2 dan nilai fungsi tujuan dapat dinaikkan tanpa batas.



Gambar 3.2

Sebagai suatu aturan umum, ketidakterbatasan dapat dinyatakan seperti berikut. Jika pada setiap iterasi elemen-elemen suatu variabel non basis adalah non positif, maka ruang solusi tak dibatasi ke arah variabel itu. Di samping itu, jika koefisien fungsi tujuan variabel itu adalah positif dalam masalah maksimisasi atau negatif dalam masalah minimisasi, maka nilai fungsi tujuan juga tak terbatas.

Selanjutnya akan ditunjukkan contoh tentang ruang solusi yang tak dibatasi sementara nilai tujuan optimum terbatas. Ingat bahwa keadaan ini juga menandai kemungkinan kesalahan dalam pembentukan model, karena ia secara tidak langsung menyatakan bahwa suatu variabel dapat dinaikkan tanpa batas.

Maksimumkan

$$Z = 6X_1 - 2X_2$$

Dengan syarat:

$$2X_1 - X_2 \leq 2$$

$$X_1 \leq 4$$

$$X_1, X_2 \geq 0$$

Tabel simpleks masalah itu adalah seperti berikut:

Tabel 3.6

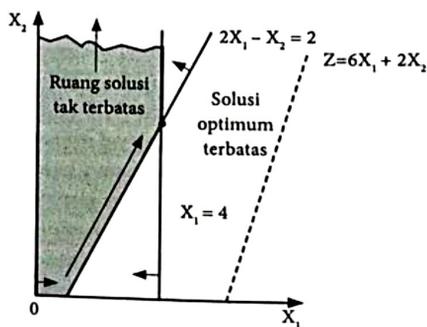
Basis	X_1	X_2	S_1	S_2	Solusi
Z	-6	2	0	0	0
S_1	2	-1	1	0	2
S_2	1	0	0	1	4

Tabel 3.7

Basis	X_1	X_2	S_1	S_2	Solusi
Z	0	-1	3	0	6
S_1	1	-1/2	1/2	0	1
S_2	0	1/2	-1/2	1	3

Tabel 3.8

Basis	X_1	X_2	S_1	S_2	Solusi
Z	0	0	2	2	12
S_1	1	0	0	1	4
S_2	0	1	-1	2	6



Gambar 3.3

Pada tabel simpleks awal ditunjukkan bahwa ruang solusi tak dibatasi ke arah X_2 . Pada Tabel 3.8 ditunjukkan bahwa nilai optimum fungsi tujuan adalah terbatas. Gambar 3.3 menunjukkan penyelesaian masalah itu dengan grafik. Jelas bahwa kemiringan fungsi tujuan adalah penting dalam menentukan apakah nilai tujuan dibatasi atau tidak.

3.3.3 Solusi Tak Layak

Jika kendala-kendala tak dapat dipenuhi secara serentak, model dikatakan tidak memiliki solusi layak. Keadaan ini tak akan terjadi jika semua kendala adalah jenis \leq (dengan konstan sisi kanan positif), karena variabel *slack* selalu memberikan solusi layak. Namun, jika digunakan kendala jenis \leq terpaksa digunakan *artificial variable*, yang tak memberikan solusi layak pada model asli. Meski *artificial variable* dapat dipaksa menjadi nol pada tabel optimum, namun ini hanya terjadi jika model itu memiliki suatu ruang solusi layak. Jika tidak, sekurang-kurangnya satu *artificial variable* akan menjadi positif pada tabel optimum. Ini merupakan petunjuk bahwa masalah itu tidak layak.

Menurut pengalaman, tidak adanya ruang solusi layak menunjukkan kemungkinan bahwa model tak dirumuskan dengan benar, karena kendala-kendala saling bertentangan. Ini juga berarti bahwa kendala-kendala tidak dipenuhi secara serentak. Contoh berikut menunjukkan kasus ruang solusi tak layak.

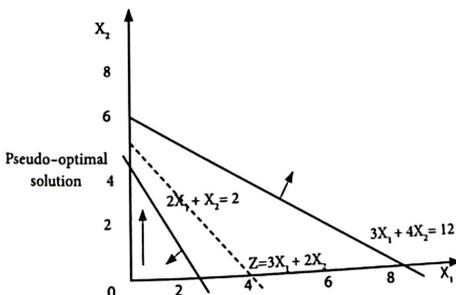
Maksimumkan

$$Z = 3X_1 + 2X_2$$

Dengan syarat:

$$\begin{aligned} 2X_1 + X_2 &\leq 2 \\ 3X_1 + 4X_2 &\geq 12 \\ X_1, X_2 &\geq 0 \end{aligned}$$

Pada tabel simpleks berikut ditunjukkan bahwa pada tabel optimum, *artificial variable* adalah positif (= 4). Ini adalah suatu indikasi bahwa ruang solusi adalah tidak layak. Gambar 3.4 menunjukkan ruang solusi tak layak. Metode simpleks yang menghasilkan *artificial variable* positif, pada intinya telah membalik arah pertidaksamaan $3X_1 + 4X_2 \geq 12$ menjadi $3X_1 + AX_2 \leq 12$. Hasil ini adalah apa yang dapat dinamakan *pseudo optimal solution*, seperti ditunjukkan Gambar 3.4.



Gambar 3.4

Tabel 3.9

Basis	X_1	X_2	S_2	S_1	A	Solusi
Z	$-3 - 3M$	$-2 - 4M$	M	0	0	$-12M$
S_1	2	1	0	1	0	2
A	3	4	-1	0	1	12

Tabel 3.10

Basis	X_1	X_2	S_2	S_1	A	Solusi
Z	$1 + 5M$	0	M	$2 + 4M$	0	$4 - 4M$
X_2	2	1	0	1	0	2
A	-5	0	-1	-4	1	4

3.4 Beberapa Masalah Teknis dalam Program Linier Metode Simpleks

Adapun beberapa masalah teknis dalam program linier metode simpleks yaitu sebagai berikut:

3.4.1 Masalah Maksimasi

Langkah-langkah penyelesaian:

1. Mengubah Fungsi Tujuan dan Batasan-batasan

- » Fungsi tujuan diubah menjadi *fungsi implisit* yaitu semuanya bergeser ke kiri.

Misalnya: Maksimumkan fungsi tujuan : $Z = 3X_1 + 5X_2$

maka menjadi : $Z - 3X_1 - 5X_2$

- » Batasan-batasan diubah menjadi kesamaan, dengan cara menambah *slack* variabel. *Slack* variabel adalah S_1, S_2, \dots, S_n . Jika hasil kegiatan yang ada mewakili X_1 dan X_2 , maka *slack* variabel dimulai dari S_1, S_2 , dan seterusnya.

Misalnya: Batasan-batasan (1) $2X_1 \leq 8$

(2) $3X_2 \leq 15$

(3) $6X_1 + 5X_2 \leq 30$

Non negatif $X_1, X_2 \geq 0$

Menjadi: Batasan-batasan

$$(1) 2X_1 + S_1 = 8$$

$$(2) 3X_2 + S_2 = 15$$

$$(3) 6X_1 + 5X_2 + S_3 = 30$$

2. Menyusun Persamaan-persamaan di dalam tabel

Metode Simpleks dalam Bentuk Simbol

Variabel Dasar	Z	X ₁	X ₂ ...	S _n ...	S ₁	X ₂ ...	S _n	NK
Z	1	-C ₁	-C ₂	-C _n	0	0...	0	0
S ₁	0	a ₁₁	a ₁₂	a _{1n}	1	0...	0	b ₁
S ₂	0	a ₂₁	a ₂₂	a _{2n}	0	1...	0	b ₂
	
	
	
S _n	0	a _{m1}	a _{m2}	a _{mn}	0	0...	1	b _m

NK adalah nilai kanan persamaan (nilai di belakang tanda =).

Variabel dasar adalah variabel yang nilainya sama dengan sisi kanan persamaan.

Variabel Dasar	Z	X ₁	X ₂	S ₁	S ₂	S ₃	NK
Z	1	-3	-5	0	0	0	0
S ₁	0	2	0	1	0	0	8
S ₂	0	0	3	0	1	0	15
S ₃	0	6	5	0	0	1	30

3. Memilih Kolom Kunci

Pilihlah kolom yang mempunyai nilai pada garis fungsi tujuan yang bernilai negatif dengan angka terbesar, dan berilah tanda segiempat pada kolom tersebut.

Variabel Dasar	Z	X ₁	X ₂	S ₁	S ₂	S ₃	NK
Z	1	-3	-5	0	0	0	0
S ₁	0	2	0	1	0	0	8
S ₂	0	0	3	0	1	0	15
S ₃	0	6	5	0	0	1	30

□ Kolom kunci

4. Memilih Baris Kunci

» Terlebih dahulu dicari indeks tiap-tiap baris, dengan rumus:

$$Indeks = \frac{\text{Nilai kolom NK}}{\text{Nilai kolom kunci}}$$

$$(0/-5, 8/0, 15/3, 30/5) = (0, \sim, 5, 6)$$

- » Pilihlah baris yang mempunyai indeks positif dengan angka terkecil.
- » Berilah tanda segiempat pada baris kunci tersebut. Nilai yang masuk dalam kolom kunci dan dalam baris kunci disebut *angka kunci*.

Variabel Dasar	Z	X_1	X_2	S_1	S_2	S_3	NK	
Z	1	-3	-5	0	0	0	0	0
S_1	0	2	0	1	0	0	8	(8/0 = ~)
S_2	0	0	3	0	1	0	15	(15/3 = 5)
S_3	0	6	5	0	0	1	30	(30/5 = 6)

□ Baris kunci, □ Angka kunci

5. Mengubah Nilai-nilai Baris Kunci, dengan cara:

Gantilah variabel dasar pada baris tersebut, dengan variabel yang terdapat di bagian atas kolom kunci.

Variabel Dasar	Z	X_1	X_2	S_1	S_2	S_3	NK
Z	1						
S_1	0						
X_2	0	0	1	0	1/3	0	5
S_3	0						

Nilai baru baris kunci: $(0/3, 3/3, 0/3, 1/3, 0/3, 15/3) = (0, 1, 0, 1/3, 0, 5)$

6. Mengubah Nilai-nilai Selain pada Baris Kunci

Dengan rumus:

Baris baru = Baris lama - (Koefisien pada kolom kunci \times Nilai baru baris kunci)

$$\begin{aligned} Z &= (-3 \ -5 \ 0 \ 0 \ 0 \ 0) - (-5) \cdot (0 \ 1 \ 0 \ 1/3 \ 0 \ 5) \\ &= (-3 \ -5 \ 0 \ 0 \ 0 \ 0) - (0 \ -5 \ 0 \ -5/3 \ 0 \ -25) \\ &= -3 \ 0 \ 0 \ 5/3 \ 0 \ 25 \end{aligned}$$

$$S_1 = \text{Baris lama, karena koefisien pada kolom kunci adalah 0}$$

$$\begin{aligned} S_3 &= (6 \ 5 \ 0 \ 0 \ 1 \ 30) - (-5) \cdot (0 \ 1 \ 0 \ 1/3 \ 0 \ 5) \\ &= (6 \ 5 \ 0 \ 0 \ 1 \ 30) - (0 \ 5 \ 0 \ 5/3 \ 0 \ 25) \\ &= 6 \ 0 \ 0 \ -5/3 \ 1 \ 5 \end{aligned}$$

Variabel Dasar	Z	X_1	X_2	S_1	S_2	S_3	NK
Z	1	-3	0	0	5/3	0	25
S_1	0	2	0	1	0	0	8
X_2	0	0	1	0	1/3	0	5
S_3	0	6	0	0	-5/3	1	5

7. Melanjutkan Perbaikan-perbaikan

Ulangi langkah ke-3 sampai dengan langkah ke-6. Perubahan baru berhenti setelah pada “baris pertama” (fungsi tujuan) tidak ada yang bernilai negatif. Berarti hasil dari Tabel tersebut sudah merupakan hasil yang optimal.

Variabel Dasar	Z	X_1	X_2	S_1	S_2	S_3	NK	
Z	1	-3	0	0	5/3	0	25	-
S_1	0	2	0	1	0	0	8	(8/2 = 4)
X_2	0	0	1	0	1/3	0	5	(5/0 = ~)
S_3	0	6	0	0	-5/3	1	5	(5/6)
Z	1							
S_1	0							
X_2	0							
X_1	0	1	0	0	-5/18	1/6	5/6	
Z	1	0	0	0	5/6	1/2	271/2	
S_1	0	0	0	1	5/9	-1/3	61/3	
X_2	0	0	1	0	1/3	1/6	5	
X_1	0	1	0	0	-5/18		5/6	

$$\begin{aligned} \text{Baris kunci yang baru} &= (6/6 \ 0/6 \ 0/6 \ -5/3/6 \ 1/6 \ 5/6) \\ &= (1 \ 0 \ 0 \ -5/18 \ 1/6 \ 5/6) \end{aligned}$$

$$\begin{aligned} Z &= (-3 \ 0 \ 0 \ 5/3 \ 0 \ 25) - (-3) \cdot (1 \ 0 \ 0 \ -5/18 \ 1/6 \ 5/6) \\ &= (-3 \ 0 \ 0 \ 5/3 \ 0 \ 25) - (-3 \ 0 \ 0 \ 15/18 \ -3/6 \ -15/6) \\ &= (0 \ 0 \ 0 \ 5/6 \ 1/2 \ 271/2) \end{aligned}$$

$$\begin{aligned} S_1 &= (2 \ 0 \ 1 \ 0 \ 0 \ 8) - (2) \cdot (1 \ 0 \ 0 \ -5/18 \ 1/6 \ 5/6) \\ &= (2 \ 0 \ 1 \ 0 \ 0 \ 8) - (2 \ 0 \ 0 \ -10/18 \ 2/6 \ 10/6) \\ &= (0 \ 0 \ 1 \ 5/9 \ -1/3 \ 6 \ 1/3) \end{aligned}$$

X_2 = Baris lama, karena koefisien pada kolom kunci adalah 0

3.4.2 Masalah Minimasi

Dalam masalah maksimasi, biasanya memiliki kendala pertidaksamaan jenis \leq . Sekarang akan diterangkan proses simpleks untuk suatu masalah minimasi yang biasanya memiliki kendala pertidaksamaan jenis \geq . Masalah minimasi menggunakan langkah-langkah yang sama seperti pada masalah maksimasi, namun ada beberapa penyesuaian yang harus dibuat bagi kendala pertidaksamaan jenis \leq maka *variable slack* ditambahkan untuk menghabiskan sumber daya yang tak digunakan dalam kendala. Namun cara ini tak dapat ditiru untuk kendala pertidaksamaan jenis dan kendala persamaan.

pada kolom solusi negatif, maka solusi yang diperoleh tak layak (melanggar kendala non negatif) dan metode simpleks tak dapat dimulai. Meskipun, coba-coba dapat diulangi lagi sampai diperoleh solusi awal layak, metode ini jelas tidak efisien dan mahal.

2. Menggunakan *Artificial Variable*

Gagasan penggunaan *artificial variable* sangat sederhana. Tambahkan suatu *artificial variable* pada sisi kiri setiap persamaan yang tidak memiliki variabel basis. Dinamakan *artificial* (sebagai lawan dari “*real decision variable*”) karena ia tidak memiliki arti nyata. *Artificial* digunakan hanya untuk memulai penyelesaian dan pada urutan selanjutnya mereka harus dijadikan nol pada solusi akhir, jika tidak, solusi yang dihasilkan akan menjadi tak layak.

Pada bentuk baku contoh di atas, *variabel slack* S_1 pada persamaan pertama adalah variabel basis. Karena pada persamaan kedua dan ketiga tidak ada variabel *slack* (variabel basis), kemudian ditambahkan *artificial variable* A_1 dan A_2 pada kedua persamaan tersebut. Untuk tetap menjamin bentuk baku, A_1 dan A_2 dibatasi pada nilai non negatif. Sehingga diperoleh *artificial system* seperti berikut:

$$\begin{array}{rcccccccc} X_1 & - & 2X_2 & + & X_3 & - & S_1 & & & = & 11 \\ -4X_1 & + & X_2 & + & 2X_3 & + & & - & S_2 & - & A_1 & = & 3 \\ -2X_1 & & & + & X_3 & & & + & & & A_2 & = & 1 \end{array}$$

Terdapat 3 persamaan dan 7 bilangan tak diketahui, sehingga solusi awal layak harus memiliki 4 ($= 7 - 3$) variabel non basis yang sama dengan nol. Jika $X_1 = X_2 = X_3 = S_2 = 0$, maka $S_1 = 11$, $A_1 = 3$ dan $A_2 = 1$. Tetapi ini bukan solusi layak karena *artificial variable* bernilai positif. Sehingga tujuan kita adalah memaksa *artificial variable* menjadi nol secepat mungkin. Ini dapat dicapai dengan dua cara. Pertama dengan *Teknik M* atau *Metode Penalty* dan yang kedua adalah *Teknik Dua Tahap*.

3.5 Soal dan Kasus

3.5.1 Soal dan Kasus 1

Ubahlah model Linear Programming berikut ke dalam bentuk baku.

Maksimumkan

$$Z = 9X_1 + 18X_2$$

Dengan syarat:

$$6X_1 + 3X_2 \leq 18$$

$$2X_1 + 2X_2 \leq 16$$

$$X_1 \text{ unrestricted}$$

$$X_2 \geq 0$$

Jawaban:

Bentuk bakunya adalah:

Maksimumkan

$$Z = 9X_1' - 9X_1'' + 18X_2 + 0S_1 + 0S_2$$

Dengan syarat:

$$6X_1' - 6X_1'' + 3X_2 + S_1 = 18$$

$$2X_1' - 2X_1'' + 2X_2 + S_2 = 16$$

$$X_1', X_1'', X_2, S_1, S_2 \geq 0$$

3.5.2 Soal dan Kasus 2

Sebuah industri kecil, memproduksi 2 macam hiasan dinding dari tripleks yang proses pengerjaannya dilakukan di 2 stasiun kerja yaitu pemotongan dan perakitan. Hiasan model A mula-mula dikerjakan di bagian pemotongan selama 5 menit kemudian dirakit selama 10 menit. Hiasan model B dikerjakan di bagian pemotongan selama 8 menit kemudian dirakit selama 8 menit. Dalam 1 hari kerja, waktu yang tersedia di bagian pemotongan 3 jam 20 menit, sedangkan di

bagian perakitan tersedia waktu 4 jam. Jika harga jual hiasan model A \$500/unit dan hiasan model B \$600/unit. Berapakah masing-masing model harus dibuat agar diperoleh total pendapatan yang optimal?

Diketahui

X_1 = Hiasan model A

X_2 = Hiasan model B

	Waktu Pengerjaan (Menit)		Waktu Tersedia Per Hari (Menit)
	X_1	X_2	
Pemotongan	5	8	200
Perakitan	10	8	240
Harga Jual/Unit (\$)	500	600	

Jawaban:

- Fungsi Tujuan:

$$\text{Maksimumkan } Z = 500X_1 + 600X_2$$

- Fungsi Kendala:

$$5X_1 + 8X_2 \leq 200$$

$$10X_1 + 8X_2 \leq 240$$

$$X_1, X_2 \geq 0$$

- Bentuk Bakunya:

$$\text{Maksimumkan } Z = 500X_1 + 600X_2 + 0S_1 + 0S_2 \text{ atau}$$

$$Z - 500X_1 - 600X_2 - 0S_1 - 0S_2$$

Fungsi Kendala:

$$5X_1 + 8X_2 + S_1 = 200$$

$$10X_1 + 8X_2 + S_2 = 240$$

$$X_1, X_2, S_1, S_2 \geq 0$$

- Solusi atau Tabel Awal Simpleks

VB	X_1	X_2	S_1	S_2	NK	Rasio
Z	-500	-600	0	0	0	
S_1	5	8	1	0	200	

Karena nilai negatif terbesar ada pada kolom X_2 , maka kolom X_2 adalah kolom pivot dan X_2 adalah variabel masuk. Rasio pembagian nilai kanan dengan kolom pivot terkecil adalah 25 bersesuaian dengan baris S_1 , maka baris S_1 adalah baris pivot dan S_2 adalah variabel keluar. Elemen pivot adalah 8.

<i>VB</i>	X_1	X_2	S_1	S_2	<i>NK</i>	Rasio
Z	-500	-600	0	0	0	
S_1	5	8	1	0	200	$200/8 = 25$
S_2	10	8	0	1	240	$240/8 = 30$

- Iterasi 1

Nilai pertama yang kita miliki adalah nilai baris pivot baru (baris X_2). Semua nilai pada baris X_2 pada tabel solusi awal dibagi dengan 8 (elemen pivot).

<i>VB</i>	X_1	X_2	S_1	S_2	<i>NK</i>	Rasio
Z	-500	-600	0	0	0	
X_2	$5/8$	1	$1/8$	0	$200/8 = 25$	
S_2						

- Perhitungan Nilai Barisnya:

Baris Z:

	-500	-600	0	0	0	
-600	$5/8$	1	$1/8$	0	25	-
:	-125	0	75	0	15.000	

Baris S_2 :

	10	8	0	1	240	
8	$5/8$	1	$1/8$	0	25	-
:	5	0	1	1	40	

Maka tabel iterasi 1 ditunjukkan tabel di bawah. Selanjutnya kita periksa apakah tabel sudah optimal atau belum. Karena nilai baris Z di bawah variabel X_1 masih negatif, maka tabel belum optimal. Kolom dan baris pivotnya ditandai pada tabel di bawah ini:

<i>VB</i>	X_1	X_2	S_1	S_2	<i>NK</i>	Rasio
Z	-125	0	75	0	15.000	
X_2	$5/8$	1	$1/8$	0	25	
S_2	5	0	1	1	40	

<i>VB</i>	X_1	X_2	S_1	S_2	<i>NK</i>	Rasio
Z	-125	0	75	0	15.000	
X_2	$5/8$	1	$1/8$	0	25	$25:5/8 = 4$
S_2	5	0	1	1	40	$40/5 = 8$

- Iterasi 2

Nilai pertama yang kita miliki adalah nilai baris pivot baru (baris X_1). Semua nilai pada baris X_1 pada tabel solusi awal dibagi dengan 5 (elemen pivot).

VB	X_1	X_2	S_1	S_2	NK	Rasio
Z						
X_2						
X_1	1	0	1/5	1/5	8	

- Perhitungan Nilai Barisnya:

Baris Z :

	-125	-125	0	75	0	15.000	
	-125	1	0	1/5	1/5	8	-
	:	0	0	125	25	16.000	

Baris X_1 :

	5/8	5/8	1	1/8	0	25	
	5/8	1	0	1/5	1/5	8	-
	:	0	3/8	0	-1/8	20	

VB	X_1	X_2	S_1	S_2	NK	Rasio
Z	0	123	100	25	16.000	
X_2	0	3/8	0	-1/8	20	
X_1	1	0	1/5	1/5	8	

- Tabel sudah optimal, sehingga perhitungan iterasi dihentikan.

Solusi optimal $X_1 = 8$, $X_2 = 20$, dan $Z = 16.000$, artinya untuk mendapatkan keuntungan maksimum sebesar \$16.000, maka perusahaan sebaiknya menghasilkan hiasan model A sebesar 8 unit dan hiasan model B sebesar 20 unit.

3.5.3 Kesimpulan

Dari pembahasan yang ada maka dapat disimpulkan bahwa metode simpleks secara konsisten dari tahap awal sampai pada tahap akhir menawarkan langkah-langkah yang terus-menerus dilakukan untuk mencapai/menjuju suatu titik penyelesaian. Metode simpleks merupakan salah satu teknik penyelesaian dalam program linier yang digunakan sebagai teknik pengambilan keputusan dalam permasalahan yang berhubungan dengan pengalokasian sumber daya secara

optimal. Metode simpleks digunakan untuk mencari nilai optimal dari program linier yang melibatkan banyak *constraint* (pembatas) dan banyak variabel (lebih dari dua variabel). Penemuan metode ini merupakan lompatan besar dalam riset operasi dan digunakan sebagai prosedur penyelesaian dari setiap program komputer. Dari berbagai metode penyelesaian program linier, metode simpleks merupakan metode yang paling ampuh dan terkenal. Metode simpleks didasarkan atas pengertian bahwa solusi optimal dari masalah program linier, jika ada, selalu dapat ditemukan disalah satu dari “solusi dasar yang berlaku”. Oleh sebab itu, dalam metode simpleks langkah pertama adalah untuk memperoleh solusi dasar yang berlaku.

3.5.4 Saran

Penulis menyadari bahwasannya penyusunan makalah ini masih terdapat banyak kesalahan dan kekurangannya, karena terbatasnya pengetahuan dan kurangnya rujukan atau referensi yang ada hubungannya dengan judul makalah ini. Oleh karena itu, penulis mengharapkan adanya kritik dan saran yang bersifat membangun dari pembaca demi kesempurnaan makalah ini dan penulisan makalah di kesempatan-kesempatan berikutnya. Semoga makalah ini dapat memberikan pengetahuan dan wawasan mendalam bagi penulis khususnya dan bagi pembaca umumnya.

Bab 4

Sistem Operasi dan Sistem Utility

4.1 Pendahuluan

Dalam dunia komputasi yang terus berkembang, sistem operasi dan sistem utility memiliki peran yang sangat penting. Keduanya merupakan fondasi dari penggunaan yang efisien dan optimal dari perangkat keras dan perangkat lunak komputer. Sistem operasi adalah perangkat lunak yang menjadi “otak” dari sebuah komputer, mengatur, mengelola, dan mengendalikan seluruh sumber daya komputer. Di sisi lain, sistem utility adalah sekumpulan perangkat lunak tambahan yang bertugas membantu pengguna dalam berbagai aspek pengelolaan komputer, mulai dari mengoptimalkan kinerja hingga menjaga keamanan dan pemulihan data.

Makalah ini akan membahas secara mendalam tentang sistem operasi dan sistem utility, mulai dari pengertian, perkembangan, fungsi-fungsi, hingga peran dan masa depan keduanya dalam era komputasi modern. Dengan memahami kedua konsep ini, kita akan dapat lebih menghargai kompleksitas dan pentingnya sistem operasi dan sistem utility dalam menjalankan dunia komputer yang begitu kompleks dan dinamis.

4.2 Pengertian Sistem Operasi dan Sistem Utility

Sistem operasi adalah perangkat lunak (software) yang menjadi perantara antara perangkat keras (*hardware*) komputer dan aplikasi yang dijalankan oleh pengguna. Sistem operasi bertanggung jawab untuk mengatur, mengelola, dan mengendalikan sumber daya komputer seperti CPU (Central Processing Unit), memori, perangkat input-output, dan perangkat keras lainnya. Sistem operasi juga menyediakan antarmuka yang memungkinkan pengguna berinteraksi dengan komputer dan menjalankan program-program aplikasi.

Sistem operasi memiliki beberapa fungsi utama, termasuk:

1. **Manajemen Sumber Daya:** Sistem operasi mengatur penggunaan sumber daya komputer seperti CPU, memori, dan perangkat keras lainnya agar dapat digunakan secara efisien oleh berbagai program.
2. **Manajemen Proses:** Sistem operasi mengendalikan eksekusi program-program (proses) dan mengatur penjadwalan tugas-tugas agar berjalan secara adil dan efisien.
3. **Manajemen Memori:** Sistem operasi mengelola alokasi memori untuk program-program yang berjalan dan memastikan penggunaan memori yang efisien.
4. **Manajemen File:** Sistem operasi memungkinkan pengguna untuk membuat, membaca, menulis, dan menghapus file di sistem komputer.
5. **Antarmuka Pengguna:** Sistem operasi menyediakan antarmuka grafis atau teks yang memungkinkan pengguna berinteraksi dengan komputer dan menjalankan program-program.

Sistem utility (perangkat lunak utilitas) adalah kelompok perangkat lunak tambahan yang dirancang untuk membantu pengguna dalam melakukan tugas-tugas tertentu yang berkaitan dengan pengelolaan dan pemeliharaan komputer. Sistem utility biasanya tidak digunakan untuk menjalankan aplikasi tertentu, tetapi untuk meningkatkan kinerja, keamanan, dan efisiensi penggunaan komputer. Beberapa contoh sistem utility meliputi:

1. **Antivirus:** Perangkat lunak antivirus digunakan untuk melindungi komputer dari serangan virus dan malware, serta melakukan pemindaian untuk mengidentifikasi dan menghapus ancaman keamanan.
2. **Utilitas Pemulihan Data:** Utility pemulihan data membantu mengembalikan data yang hilang atau terhapus secara tidak sengaja dari penyimpanan komputer.
3. **Program Pengarsipan File:** Utility pengarsipan file digunakan untuk mengompres dan mengarsipkan file, menghemat ruang penyimpanan, dan memudahkan pengiriman file.

4. **Optimasi Sistem:** Utility optimasi sistem membantu mempercepat kinerja komputer dengan membersihkan file sampah, memperbaiki registri, dan mengoptimalkan pengaturan.
5. **Firewall:** Perangkat lunak firewall melindungi komputer dari akses yang tidak sah dan mengatur lalu lintas jaringan untuk menjaga keamanan.

Sistem utility berperan penting dalam menjaga kesehatan dan kinerja komputer serta meningkatkan pengalaman pengguna. Mereka adalah alat tambahan yang mendukung fungsi sistem operasi dalam mengelola komputer.

4.3 Pengertian Perangkat Lunak Utility

Perangkat lunak utility, juga dikenal sebagai utilitas perangkat lunak, adalah jenis perangkat lunak yang dirancang untuk membantu pengguna dalam melakukan tugas-tugas tertentu yang berkaitan dengan pengelolaan dan pemeliharaan komputer. Perangkat lunak utility biasanya tidak digunakan untuk menjalankan aplikasi utama atau program kreatif, tetapi bertujuan untuk meningkatkan kinerja, keamanan, keandalan, atau efisiensi penggunaan komputer.

Berikut adalah beberapa contoh perangkat lunak utility dan tugas-tugas yang mereka tangani:

1. **Antivirus:** Perangkat lunak antivirus adalah utility yang dirancang untuk melindungi komputer dari serangan virus, malware, spyware, dan ancaman keamanan lainnya. Mereka melakukan pemindaian sistem untuk mendeteksi dan menghapus ancaman potensial.
2. **Utilitas Pemulihan Data:** Utilitas pemulihan data membantu mengembalikan data yang hilang atau terhapus secara tidak sengaja dari perangkat penyimpanan, seperti hard drive atau kartu memori.
3. **Program Pengarsipan File:** Program pengarsipan file atau utilitas kompresi file membantu mengompres atau mengarsipkan file dan folder, yang mengurangi ukuran file dan memudahkan pengiriman atau penyimpanan.
4. **Utilitas Pembersihan Disk:** Utilitas ini membersihkan file sementara, file sampah, dan cache yang tidak diperlukan pada komputer, mengoptimalkan ruang penyimpanan dan kinerja.

5. **Optimasi Sistem:** Utility optimasi sistem membantu memperbaiki dan mengoptimalkan kinerja komputer dengan membersihkan registri sistem, mengatur pengaturan sistem, dan mengidentifikasi masalah kinerja.
6. **Firewall:** Perangkat lunak firewall adalah utility yang melindungi komputer dari akses yang tidak sah dan memantau lalu lintas jaringan untuk mengamankan koneksi internet.
7. **Utilitas Manajemen Partisi:** Utility ini memungkinkan pengguna untuk membuat, mengubah, menghapus, atau mengelola partisi pada hard drive mereka, yang penting untuk pengorganisasian dan penyimpanan data.
8. **Utilitas Pengelolaan Driver:** Utility ini membantu dalam mengelola driver perangkat keras, termasuk pembaruan, penghapusan, dan pemulihan driver yang dibutuhkan.
9. **Utilitas Keamanan Kata Sandi:** Utility keamanan kata sandi membantu mengelola dan menyimpan kata sandi dengan aman serta menghasilkan kata sandi yang kuat.

Perangkat lunak utility adalah alat yang sangat berharga bagi pengguna komputer karena mereka membantu menjaga kinerja dan keamanan sistem, serta menyederhanakan tugas pemeliharaan yang kompleks. Dengan beragam tugas yang dapat mereka tangani, perangkat lunak utility menjadi bagian integral dari pengalaman pengguna yang modern.

4.4 Perkembangan Sistem Operasi

Perkembangan Sistem Operasi (Operating System Development) merupakan salah satu aspek yang sangat penting dalam dunia komputasi. Seiring dengan perkembangan teknologi komputer dan kebutuhan pengguna yang semakin kompleks, sistem operasi telah mengalami perubahan dan peningkatan yang signifikan sejak awal komputasi hingga era modern. Berikut adalah ikhtisar perkembangan sistem operasi:

1. **Sistem Operasi Awal (1940–1950-an):** Pada awal perkembangan komputer, sistem operasi sangat sederhana dan terbatas. Salah satu sistem operasi pertama adalah EDSAC (Electronic Delay Storage Automatic Calculator)

yang dikembangkan pada tahun 1949. Sistem operasi awal ini umumnya berfokus pada pengelolaan perangkat keras dasar dan mengatur eksekusi program dengan sangat terbatas.

2. **Era Sistem Operasi Batch (1950–1960-an):** Dalam era ini, sistem operasi batch menjadi populer. Contohnya adalah IBM's OS/360. Sistem operasi batch memungkinkan sekelompok tugas atau pekerjaan untuk dijalankan secara bersamaan tanpa intervensi pengguna. Namun, interaksi langsung dengan komputer masih sangat terbatas.
3. **Munculnya Sistem Operasi Time-Sharing (1960–1970-an):** Era ini ditandai oleh pengembangan sistem operasi yang mendukung time-sharing, seperti CTSS (Compatible Time-Sharing System) dan MULTICS. Time-sharing memungkinkan beberapa pengguna untuk berbagi komputer secara bersamaan dan memiliki akses interaktif ke sistem.
4. **Era Sistem Operasi Mikrokomputer (1970–1980-an):** Dalam dekade ini, komputer mikro (microcomputers) mulai populer, dan sistem operasi seperti CP/M, MS-DOS, dan Apple DOS muncul. Sistem operasi ini dirancang untuk komputer pribadi (personal computer) yang lebih terjangkau.
5. **Perkembangan Sistem Operasi Grafis (1980–1990-an):** Munculnya sistem operasi berbasis GUI (Graphical User Interface) seperti Apple Macintosh, Microsoft Windows, dan sistem berbasis X Window (untuk komputer Unix/Linux) membawa revolusi dalam cara pengguna berinteraksi dengan komputer. Antarmuka yang lebih intuitif dengan penggunaan mouse dan ikon memungkinkan pengguna untuk menjalankan program dengan lebih mudah.
6. **Era Sistem Operasi Jaringan (1990–2000-an):** Seiring dengan perkembangan Internet, sistem operasi semakin fokus pada dukungan jaringan. Windows NT dan Unix/Linux semakin populer di lingkungan bisnis dan server.
7. **Perkembangan Sistem Operasi Mobile (2000–sekarang):** Dengan munculnya perangkat mobile, seperti smartphone dan tablet, sistem operasi mobile seperti Android dan iOS menjadi dominan. Mereka dirancang khusus untuk perangkat seluler dan memberikan penggunaan yang lebih *mobile-friendly*.

8. **Sistem Operasi Cloud dan Virtualisasi (2010-an - sekarang):** Sistem operasi semakin terintegrasi dengan teknologi cloud computing dan virtualisasi. Cloud OS seperti Windows Azure dan Amazon Web Services (AWS) memungkinkan pengelolaan sumber daya di cloud secara efisien. Virtualisasi juga memungkinkan berjalannya beberapa sistem operasi di dalam satu perangkat keras fisik.
9. **Sistem Operasi Masa Depan:** Masa depan sistem operasi diperkirakan akan lebih terhubung dengan teknologi cloud, kecerdasan buatan (AI), Internet of Things (IoT), dan komputasi kuantum. Sistem operasi akan terus berkembang untuk mendukung perangkat dan aplikasi yang semakin kompleks dan terhubung.

Perkembangan sistem operasi selalu berjalan seiring dengan kemajuan teknologi komputer dan kebutuhan pengguna. Sistem operasi yang lebih canggih dan adaptif terus dibuat untuk mendukung berbagai perangkat dan aplikasi dalam ekosistem komputasi yang terus berkembang.

4.5 Fungsi-fungsi Sistem Operasi

Sistem operasi (*Operating System*) memiliki berbagai fungsi penting yang membantu mengatur dan mengendalikan sumber daya komputer serta memberikan lingkungan untuk menjalankan program-program. Berikut adalah beberapa fungsi utama dari sistem operasi:

1. **Manajemen Sumber Daya:** Sistem operasi bertanggung jawab untuk mengatur dan mengelola seluruh sumber daya perangkat keras, termasuk CPU (Central Processing Unit), memori, perangkat input-output (I/O), dan perangkat keras lainnya. Hal ini dilakukan agar sumber daya tersebut dapat digunakan secara efisien oleh berbagai program.
2. **Manajemen Proses:** Sistem operasi mengelola eksekusi program-program (proses) pada komputer. Ini termasuk penjadwalan tugas-tugas, pemantauan status proses, serta alokasi sumber daya seperti CPU dan memori.
3. **Manajemen Memori:** Sistem operasi mengatur dan mengelola alokasi memori untuk program-program yang berjalan. Ini melibatkan alokasi, dealokasi, dan penggunaan memori fisik dan virtual untuk memastikan penggunaan memori yang efisien.

4. **Manajemen File:** Sistem operasi memungkinkan pengguna untuk membuat, membaca, menulis, dan menghapus file dan direktori. Ini juga mencakup pengaturan hak akses dan struktur direktori.
5. **Antarmuka Pengguna:** Sistem operasi menyediakan antarmuka yang memungkinkan pengguna berinteraksi dengan komputer. Antarmuka ini bisa berupa antarmuka berbasis teks (command-line interface) atau antarmuka berbasis grafis (graphical user interface - GUI) yang lebih intuitif.
6. **Manajemen Perangkat Keras:** Sistem operasi mengendalikan dan mengatur interaksi dengan perangkat keras seperti keyboard, mouse, layar, printer, dan perangkat lainnya. Ini memastikan bahwa perangkat keras dapat berkomunikasi dengan perangkat lunak.
7. **Manajemen Jaringan:** Dalam lingkungan jaringan, sistem operasi menyediakan dukungan untuk koneksi jaringan, mengelola alamat IP, dan mengatur lalu lintas data antara komputer dalam jaringan.
8. **Keamanan:** Sistem operasi memiliki peran penting dalam menjaga keamanan sistem. Ini melibatkan pengaturan hak akses, enkripsi data, perlindungan terhadap serangan keamanan seperti virus dan malware, serta manajemen kata sandi.
9. **Manajemen Energi:** Pada perangkat mobile dan laptop, sistem operasi dapat mengelola konsumsi daya agar lebih efisien, termasuk mengatur mode tidur (sleep mode) dan manajemen daya baterai.
10. **Pemulihan dan Keandalan:** Sistem operasi memiliki kemampuan untuk mendeteksi dan menangani kesalahan (error handling) serta memberikan mekanisme pemulihan dalam situasi yang tidak terduga.
11. **Manajemen Virtualisasi:** Sistem operasi modern mendukung teknologi virtualisasi, yang memungkinkan berjalan beberapa sistem operasi atau lingkungan virtual di dalam satu komputer fisik.
12. **Manajemen Kinerja:** Sistem operasi memantau kinerja komputer dan dapat memberikan alat untuk menganalisis dan memperbaiki kinerja sistem.
13. **Manajemen Layanan:** Dalam sistem operasi berbasis server, sistem operasi dapat mengelola dan menyediakan layanan seperti server web, server basis data, dan server jaringan lainnya.

Fungsi-fungsi ini membuat sistem operasi menjadi jantung dari pengoperasian komputer modern. Mereka memungkinkan pengguna untuk menjalankan program, mengakses sumber daya, dan berinteraksi dengan komputer dengan cara yang efisien dan aman.

4.6 Fungsi-fungsi Utility

Perangkat lunak utility (software utility) adalah kelompok perangkat lunak tambahan yang dirancang untuk membantu pengguna dalam melakukan tugas-tugas tertentu yang berkaitan dengan pengelolaan dan pemeliharaan komputer. Fungsi-fungsi utility ini bertujuan untuk meningkatkan kinerja, keamanan, keandalan, dan efisiensi penggunaan komputer. Berikut adalah beberapa fungsi umum dari perangkat lunak utility:

1. **Keamanan:**

- » **Antivirus:** Melindungi komputer dari serangan virus, malware, dan spyware dengan pemindaian dan deteksi ancaman.
- » **Firewall:** Melindungi komputer dari akses yang tidak sah dan mengatur lalu lintas jaringan.
- » **Perangkat Lunak Enkripsi:** Mengamankan data dengan mengenkripsi file atau folder tertentu.
- » **Pemantauan Keamanan:** Memantau aktivitas sistem untuk mendeteksi aktivitas mencurigakan atau ancaman keamanan.

2. **Pemeliharaan Sistem:**

- » **Utilitas Pembersihan Disk:** Menghapus file sementara, file sampah, dan cache untuk mengoptimalkan ruang penyimpanan dan kinerja.
- » **Pengoptimalan Registri:** Membersihkan dan mengoptimalkan registri sistem untuk mencegah kesalahan dan mempercepat komputer.
- » **Pengelolaan Driver:** Memeriksa, mengupdate, atau menghapus driver perangkat keras yang tidak diperlukan.
- » **Pemantauan Kesehatan Hardware:** Memantau kondisi perangkat keras komputer seperti suhu CPU, kecepatan kipas, dan kesehatan hard drive.

3. Pemulihan Data:

- » **Utilitas Pemulihan Data:** Mengembalikan data yang hilang atau terhapus secara tidak sengaja dari perangkat penyimpanan.
- » **Backup dan Restore:** Mengelola cadangan data dan memungkinkan pengguna untuk memulihkan data jika terjadi kegagalan.

4. Pengarsipan dan Kompresi:

- » **Program Pengarsipan File:** Mengompres dan mengarsipkan file untuk menghemat ruang penyimpanan dan memudahkan pengiriman file.
- » **Utilitas Kompresi:** Mengompresi file dan folder untuk mengurangi ukuran dan menghemat ruang penyimpanan.

5. Optimasi Sistem:

- » **Utilitas Optimasi:** Meningkatkan kinerja komputer dengan menghapus file yang tidak diperlukan, memperbaiki konfigurasi sistem, dan mengoptimalkan pengaturan.
- » **Defragmentasi Hard Drive:** Mengatur ulang data pada hard drive untuk mengoptimalkan akses dan kinerja.

6. Manajemen File:

- » **Utilitas Manajemen File:** Menyediakan fitur tambahan untuk mengelola, menyalin, memindahkan, atau menggabungkan file dan folder.
- » **Pencarian dan Pemfilteran:** Membantu dalam pencarian dan filter file berdasarkan kriteria tertentu.

7. Keamanan Kata Sandi:

- » **Manajer Kata Sandi:** Menyimpan dan mengelola kata sandi secara aman serta menghasilkan kata sandi yang kuat.

8. Jaringan dan Internet:

- » **Utilitas Jaringan:** Memantau koneksi jaringan, menguji kecepatan internet, dan melakukan konfigurasi jaringan.
- » **Manajer Download:** Meningkatkan pengelolaan unduhan dari internet.

9. **Virtualisasi dan Manajemen Sistem:**

- » ***Virtual Machine Management:*** Memungkinkan pembuatan, konfigurasi, dan pengelolaan mesin virtual di komputer.
- » **Utilitas Manajemen Sistem:** Memantau dan mengoptimalkan kinerja sistem secara keseluruhan.

10. **Pengarsipan dan Keamanan Email:**

- » **Program Keamanan Email:** Melindungi email dari ancaman malware dan phishing.
- » **Utilitas Pencadangan Email:** Membuat cadangan email untuk melindungi data penting.

Fungsi-fungsi utility ini membantu pengguna untuk menjaga kinerja, keamanan, dan keandalan sistem komputer mereka. Mereka merupakan alat yang sangat berguna dalam pemeliharaan dan pengelolaan komputer modern.

4.7 Jenis-jenis Perangkat Lunak Sistem Operasi

Sistem operasi adalah perangkat lunak yang sangat penting dalam pengelolaan dan pengoperasian komputer. Terdapat beberapa jenis sistem operasi yang berbeda, yang biasanya digunakan untuk tujuan tertentu atau pada berbagai jenis perangkat keras. Berikut adalah beberapa jenis utama perangkat lunak sistem operasi:

1. **Sistem Operasi Desktop (*Desktop Operating Systems*):**

- » **Microsoft Windows:** Sistem operasi ini sangat umum digunakan pada komputer desktop dan laptop. Versi populer termasuk Windows 10 dan Windows 11.
- » **macOS:** Digunakan pada komputer Macintosh buatan Apple. Ini mencakup versi-versi seperti macOS Big Sur dan macOS Catalina.
- » **Linux:** Sistem operasi open source yang tersedia dalam berbagai distribusi seperti Ubuntu, CentOS, dan Fedora. Linux digunakan pada komputer desktop, server, dan perangkat lainnya.

2. **Sistem Operasi Server (*Server Operating Systems*):**

- » **Windows Server:** Versi khusus dari sistem operasi Windows yang dirancang untuk digunakan pada server. Ini mencakup Windows Server 2019 dan versi-versi sebelumnya.
- » **Linux Server:** Distribusi Linux yang dioptimalkan untuk digunakan pada server, seperti Ubuntu Server, CentOS, dan Debian.
- » **UNIX:** Sistem operasi UNIX adalah keluarga sistem operasi yang digunakan secara luas pada server dan sistem komputasi besar.

3. **Sistem Operasi Mobile (*Mobile Operating Systems*):**

- » **Android:** Sistem operasi mobile yang dikembangkan oleh Google. Digunakan pada sebagian besar smartphone dan tablet di seluruh dunia.
- » **iOS:** Sistem operasi mobile yang digunakan eksklusif pada produk-produk Apple, seperti iPhone dan iPad.

4. **Sistem Operasi Real-Time (*Real-Time Operating Systems - RTOS*):**

- » RTOS dirancang untuk aplikasi yang memerlukan respon waktu nyata, seperti kendali industri, kendaraan otonom, dan peralatan medis.

Contoh RTOS termasuk FreeRTOS, QNX, dan VxWorks.

5. **Sistem Operasi Tertanam (*Embedded Operating Systems*):**

Digunakan dalam perangkat tertanam seperti mesin cuci, peralatan rumah tangga, kamera digital, dan peralatan medis.

Contoh sistem operasi tertanam termasuk Embedded Linux, Windows Embedded, dan FreeRTOS.

6. **Sistem Operasi Jaringan (*Network Operating Systems*):**

Digunakan pada server jaringan untuk mengelola dan mengkoordinasikan sumber daya jaringan seperti file, printer, dan aplikasi.

Contoh sistem operasi jaringan meliputi Windows Server (untuk lingkungan Windows) dan Linux (untuk lingkungan open source).

7. **Sistem Operasi Cloud (*Cloud Operating Systems*):**

Dirancang untuk mengelola infrastruktur komputasi cloud dan sumber daya cloud seperti server virtual, penyimpanan, dan jaringan.

Contoh sistem operasi cloud termasuk Amazon Web Services (AWS) Elastic Compute Cloud (EC2) dan Google Cloud Platform (GCP).

8. **Sistem Operasi Komputer Kuantum (*Quantum Computing Operating Systems*):**

Dikembangkan untuk mengelola dan menjalankan komputer kuantum yang masih dalam tahap eksperimen dan pengembangan.

Setiap jenis sistem operasi memiliki karakteristik uniknya sendiri dan ditujukan untuk mengatasi kebutuhan khusus dalam berbagai konteks komputasi. Jenis sistem operasi yang digunakan akan sangat bergantung pada perangkat keras yang digunakan dan tujuan penggunaannya.

4.8 Jenis-jenis Perangkat Lunak Utility

Perangkat lunak utility (software utility) mencakup berbagai jenis alat dan aplikasi yang dirancang untuk membantu pengguna dalam menjalankan tugas-tugas tertentu yang berkaitan dengan pengelolaan dan pemeliharaan komputer. Berikut adalah beberapa jenis perangkat lunak utility yang umum digunakan:

1. **Antivirus:**

Fungsi: Melindungi komputer dari serangan virus, malware, spyware, dan ancaman keamanan lainnya dengan mendeteksi dan menghapus ancaman.

Contoh: McAfee, Norton, Avast, AVG, Malwarebytes.

2. **Firewall:**

Fungsi: Melindungi komputer dari akses yang tidak sah dan mengontrol lalu lintas jaringan untuk menjaga keamanan.

Contoh: Windows Firewall, ZoneAlarm, Norton Firewall.

3. **Utilitas Pembersihan Disk:**

Fungsi: Menghapus file sementara, file sampah, dan cache untuk mengoptimalkan ruang penyimpanan dan kinerja komputer.

Contoh: CCleaner, Wise Disk Cleaner, CleanMyPC.

4. **Utilitas Pemulihan Data:**

Fungsi: Mengembalikan data yang hilang atau terhapus secara tidak sengaja dari perangkat penyimpanan.

Contoh: Recuva, Stellar Data Recovery, EaseUS Data Recovery.

5. **Utilitas Pengarsipan File:**

Fungsi: Mengompresi dan mengarsipkan file dan folder untuk menghemat ruang penyimpanan dan memudahkan pengiriman.

Contoh: WinZip, 7-Zip, WinRAR.

6. **Utilitas Pemeliharaan Sistem:**

Fungsi: Memelihara dan mengoptimalkan kinerja komputer dengan membersihkan registri, memperbaiki kesalahan sistem, dan mengatur pengaturan.

Contoh: Glary Utilities, Advanced SystemCare, System Mechanic.

7. **Manajer Kata Sandi:**

Fungsi: Menyimpan dan mengelola kata sandi dengan aman serta menghasilkan kata sandi yang kuat.

Contoh: LastPass, Dashlane, 1Password.

8. **Utilitas Manajemen Driver:**

Fungsi: Mengelola driver perangkat keras, termasuk pembaruan, penghapusan, dan pemulihan driver yang dibutuhkan.

Contoh: Driver Booster, DriverMax, Snappy Driver Installer.

9. **Utilitas Pemulihan dan Cadangan Sistem:**

Fungsi: Membuat cadangan sistem operasi dan data, serta memungkinkan pemulihan sistem dalam situasi darurat.

Contoh: Acronis True Image, Macrium Reflect, EaseUS Todo Backup.

10. **Utilitas Keamanan Jaringan:**

Fungsi: Melindungi komputer dari serangan jaringan, mengatur pengaturan keamanan, dan memantau lalu lintas jaringan.

Contoh: Wireshark, Norton Internet Security, Bitdefender.

11. **Utilitas Manajemen Partisi:**

Fungsi: Membantu pengguna dalam membuat, mengubah, menghapus, atau mengelola partisi pada hard drive.

Contoh: EaseUS Partition Master, MiniTool Partition Wizard, Paragon Partition Manager.

12. Utilitas Pemantauan Kesehatan Hardware:

Fungsi: Memantau suhu CPU, kecepatan kipas, kesehatan hard drive, dan perangkat keras lainnya.

Contoh: HWMonitor, CrystalDiskInfo, Speccy.

13. Manajer Unduhan:

Fungsi: Mengelola unduhan dari internet dengan cepat dan efisien.

Contoh: Internet Download Manager (IDM), Free Download Manager, EagleGet.

14. Utilitas Pengarsipan Email:

Fungsi: Melindungi email dari ancaman keamanan dan memungkinkan pencadangan email.

Contoh: Norton AntiSpam, SpamSieve (untuk Apple Mail).

15. Utilitas Jaringan:

Fungsi: Memantau dan mengelola koneksi jaringan, mendiagnosis masalah, dan mengatur konfigurasi jaringan.

Contoh: NetBalancer, Wi-Fi Analyzer, PingPlotter.

Setiap jenis perangkat lunak utility memiliki peran khususnya sendiri dalam menjaga kinerja, keamanan, dan efisiensi komputer. Pemilihan perangkat lunak utility yang tepat dapat membantu pengguna dalam memelihara dan mengoptimalkan sistem komputer mereka.

4.9 Struktur pada Program Utility

Struktur pada program utility atau perangkat lunak utilitas (utility software) dapat bervariasi tergantung pada tujuan dan fungsi khususnya. Namun, ada beberapa komponen umum yang dapat ditemukan dalam struktur program utility. Berikut adalah komponen umum dalam struktur program utility:

1. Antarmuka Pengguna (*User Interface*):

Program utility biasanya memiliki antarmuka pengguna (UI) yang memungkinkan pengguna berinteraksi dengan program tersebut. Antarmuka ini dapat berupa antarmuka berbasis teks (command-line interface) atau antarmuka berbasis grafis (graphical user interface - GUI), tergantung pada jenis utility dan preferensi pengguna.

2. **Modul Utama (*Main Module*):**

Modul utama adalah bagian inti dari program utility yang menjalankan fungsi utama. Ini adalah bagian program yang melakukan tugas-tugas yang diinginkan oleh pengguna.

3. **Pengaturan (*Settings*) dan Preferensi (*Preferences*):**

Sebagian besar program utility menyediakan opsi pengaturan dan preferensi yang memungkinkan pengguna untuk menyesuaikan perilaku program sesuai dengan kebutuhan mereka. Ini dapat mencakup pengaturan seperti frekuensi pemindaian, lokasi penyimpanan, atau konfigurasi jaringan.

4. **Log dan Catatan (*Logs and Records*):**

Beberapa program utility dapat menyimpan catatan atau log aktivitas yang dilakukan oleh program. Log ini dapat berguna untuk mendiagnosis masalah atau melacak perubahan yang telah dilakukan oleh program.

5. **Algoritma dan Logika Bisnis (*Algorithms and Business Logic*):**

Ini adalah bagian yang berisi algoritma dan logika yang mendefinisikan cara program utility berfungsi. Ini mencakup semua proses pemrosesan yang dilakukan oleh program untuk mencapai tujuannya.

6. **Pustaka (*Libraries*) atau Fungsi Bawaan (*Built-in Functions*):**

Program utility sering menggunakan pustaka atau fungsi bawaan untuk melakukan tugas-tugas tertentu. Ini bisa berupa fungsi-fungsi yang telah dikembangkan sebelumnya dan diambil dari pustaka yang sudah ada.

7. **Pemantauan dan Pelaporan (*Monitoring and Reporting*):**

Program utility mungkin memiliki kemampuan pemantauan yang memungkinkan pengguna untuk melacak kinerja atau aktivitas sistem. Mereka juga dapat menghasilkan laporan atau notifikasi jika terjadi perubahan atau masalah.

8. **Fungsi Tambahan (*Additional Functions*):**

Beberapa program utility mungkin memiliki fungsi tambahan yang tidak terkait langsung dengan fungsi utama tetapi dapat berguna. Ini bisa mencakup dukungan untuk berbagai format file, integrasi dengan layanan cloud, atau bahkan integrasi dengan perangkat keras khusus.

9. **Dokumentasi (*Documentation*):**

Dokumentasi yang baik adalah bagian penting dari program utility. Ini termasuk panduan pengguna, petunjuk instalasi, dan informasi yang menjelaskan cara menggunakan program dan fungsi-fungsi yang disediakan.

10. **Manajemen Kesalahan (*Error Handling*):**

Program utility biasanya memiliki mekanisme manajemen kesalahan yang memungkinkan program untuk mengatasi situasi yang tidak diinginkan atau kesalahan yang terjadi selama operasi.

Struktur ini dapat bervariasi tergantung pada kompleksitas program utility dan tujuan pengembangannya. Dalam pengembangan program utility yang lebih besar, struktur ini dapat dibagi menjadi komponen yang lebih kecil atau modul yang berbeda untuk meningkatkan keterbacaan dan pemeliharaan kode.

4.10 Cara Kerja Sistem Operasi

Cara kerja sistem operasi (Operating System) dapat diuraikan menjadi beberapa tahap dan konsep dasar yang mendasarinya. Berikut adalah cara umum bagaimana sistem operasi bekerja:

1. **Bootstrapping (*Bootstrap Process*):**

Ketika komputer dinyalakan, sistem operasi harus dimuat ke dalam memori utama (RAM) dari perangkat penyimpanan permanen (seperti hard drive atau SSD). Proses ini disebut “bootstrapping” atau “booting.” Biasanya, BIOS (Basic Input/Output System) atau UEFI (Unified Extensible Firmware Interface) akan memulai proses ini dan memuat bootloader pertama ke dalam memori.

2. **Inisialisasi (*Initialization*):**

Setelah bootloader dimuat, sistem operasi menginisialisasi perangkat keras dan sumber daya komputer seperti CPU, memori, perangkat input-output (I/O), dan perangkat penyimpanan. Ini juga melibatkan pengecekan dan pengaturan awal perangkat keras untuk memastikan semuanya berfungsi dengan baik.

3. **Penjadwalan Proses (*Process Scheduling*):**

Sistem operasi bertanggung jawab untuk mengatur dan mengelola eksekusi program-program yang berjalan pada komputer. Ini melibatkan penjadwalan proses, di mana sistem operasi memutuskan urutan dan waktu pelaksanaan setiap program atau tugas.

4. **Manajemen Memori (*Memory Management*):**

Sistem operasi mengelola alokasi memori untuk program-program yang berjalan. Ini mencakup pemantauan penggunaan memori, alokasi memori baru, dan penghapusan memori yang tidak lagi diperlukan.

5. **Manajemen File (*File Management*):**

Sistem operasi menyediakan layanan untuk membuat, membaca, menulis, menghapus, dan mengatur file dan direktori pada perangkat penyimpanan. Ini termasuk pengelolaan hak akses dan struktur direktori.

6. **Manajemen Input-Output (*I/O Management*):**

Sistem operasi mengendalikan perangkat input-output seperti keyboard, mouse, monitor, printer, dan perangkat lainnya. Ini memastikan data dapat masuk dan keluar dari komputer dengan benar.

7. **Manajemen Jaringan (*Network Management*):**

Jika komputer terhubung ke jaringan, sistem operasi juga bertanggung jawab untuk mengelola koneksi jaringan, pengaturan alamat IP, dan mentransmisikan data melalui jaringan.

8. **Manajemen Perangkat Keras (*Hardware Management*):**

Sistem operasi mengelola interaksi dengan perangkat keras komputer, termasuk driver perangkat keras, penggunaan CPU, dan kontrol perangkat keras lainnya.

9. **Keamanan (*Security*):**

Sistem operasi memiliki fitur keamanan yang berfungsi untuk melindungi sistem dan data dari akses yang tidak sah dan serangan keamanan. Ini termasuk pengaturan hak akses, enkripsi data, dan perangkat lunak antivirus.

10. **Manajemen Daya (*Power Management*):**

Pada perangkat mobile dan laptop, sistem operasi dapat mengelola konsumsi daya untuk menghemat baterai, seperti mengaktifkan mode tidur (sleep mode) atau menonaktifkan perangkat keras yang tidak digunakan.

11. **Manajemen Error (*Error Handling*):**

Sistem operasi memiliki mekanisme untuk mendeteksi dan menangani kesalahan, seperti crash aplikasi atau masalah perangkat keras.

12. **Pemantauan dan Pelaporan (*Monitoring and Reporting*):**

Sistem operasi dapat memantau kinerja sistem dan menghasilkan laporan yang berguna untuk pemantauan dan pemecahan masalah.

Sistem operasi menjalankan tugas-tugas ini secara terus-menerus dan berinteraksi dengan pengguna dan program-program yang berjalan. Cara kerja sistem operasi yang efisien dan andal sangat penting untuk menjaga stabilitas dan kinerja komputer.

4.11 Sistem Operasi Masa Depan

Sistem operasi terus mengalami perkembangan dan evolusi seiring dengan kemajuan teknologi komputer dan perubahan dalam kebutuhan pengguna. Berikut adalah beberapa tren dan perkembangan yang mungkin menjadi bagian dari sistem operasi masa depan:

1. **Kemampuan Kecerdasan Buatan (*Artificial Intelligence - AI*):**

Sistem operasi masa depan mungkin akan mengintegrasikan teknologi kecerdasan buatan untuk meningkatkan pengalaman pengguna. Ini bisa termasuk kemampuan untuk mengenali pola penggunaan, memberikan rekomendasi, dan melakukan tugas-tugas rutin secara otomatis.

2. **Komputasi Kuantum (*Quantum Computing*):**

Dengan kemajuan dalam teknologi komputasi kuantum, sistem operasi masa depan mungkin perlu mendukung lingkungan komputasi kuantum. Ini akan memungkinkan pemrosesan data yang jauh lebih cepat dan kompleks daripada komputer klasik.

3. **Keamanan yang Lebih Tinggi:**

Keamanan tetap menjadi perhatian utama, dan sistem operasi masa depan akan terus meningkatkan lapisan keamanan. Ini termasuk enkripsi yang lebih kuat, deteksi ancaman otomatis, dan manajemen hak akses yang lebih canggih.

4. **Pengelolaan Multicloud:**

Dalam lingkungan bisnis yang semakin mengadopsi komputasi awan (cloud computing), sistem operasi mungkin akan dirancang untuk mengelola dan mengintegrasikan berbagai layanan cloud dari berbagai penyedia cloud.

5. **Ketersediaan yang Lebih Luas:**

Sistem operasi masa depan akan mendukung berbagai jenis perangkat, termasuk perangkat cerdas (smart devices) dalam Internet of Things (IoT), perangkat wearable, kendaraan otonom, dan lain-lain. Ini akan menciptakan ekosistem yang lebih terintegrasi.

6. **Peningkatan Efisiensi Energi:**

Pada perangkat mobile dan laptop, sistem operasi akan semakin berfokus pada efisiensi energi untuk meningkatkan masa pakai baterai dan mengurangi dampak lingkungan.

7. **Virtualisasi yang Lebih Lanjut:**

Sistem operasi akan terus mendukung virtualisasi dan kontainerisasi, memungkinkan pengguna untuk menjalankan banyak lingkungan berbeda di dalam satu perangkat keras.

8. **Pembaruan Perangkat Lunak yang Lebih Aman dan Lancar:**

Pembaruan sistem operasi mungkin akan menjadi lebih otomatis dan lebih aman, dengan lebih sedikit gangguan pada pengguna. Pembaruan berbasis kontainer dan pembaruan aplikasi yang lebih lancar dapat menjadi bagian dari tren ini.

9. **Integrasi Realitas Virtual (*Virtual Reality - VR*) dan *Augmented Reality (AR)*:**

Sistem operasi mungkin akan mendukung integrasi yang lebih kuat dengan teknologi VR dan AR untuk pengalaman pengguna yang lebih immersif dan produktif.

10. ***Open Source* dan Kolaborasi:**

Open source dan kolaborasi antara berbagai komunitas pengembang mungkin akan menjadi lebih penting dalam pengembangan sistem operasi masa depan. Ini dapat memungkinkan inovasi yang lebih cepat dan beragam.

Sistem operasi masa depan akan terus berubah dan beradaptasi dengan perkembangan teknologi dan kebutuhan pengguna. Peran sistem operasi dalam pengoperasian perangkat komputer dan pengelolaan sumber daya akan tetap sentral, tetapi akan berintegrasi dengan teknologi terbaru untuk meningkatkan produktivitas, keamanan, dan kenyamanan pengguna.

4.12 Soal dan Kasus

Contoh Soal:

1. Apa perbedaan utama antara sistem operasi desktop dan sistem operasi server?
2. Jelaskan apa yang dimaksud dengan manajemen memori dalam sistem operasi.
3. Apa peran utama dari perangkat lunak antivirus dalam sistem operasi?
4. Bagaimana sistem operasi mengelola penjadwalan proses?
5. Mengapa manajemen file penting dalam sistem operasi? Sebutkan beberapa operasi file yang umum digunakan.

Jawab:

1. Perbedaan utama antara Server dengan komputer desktop sebetulnya bukan pada hardware tapi lebih kepada softwarenya. Server umumnya menjalankan sistem operasi yang dirancang untuk digunakan di server. Server akan menjalankan aplikasi yang khusus untuk menjalankan tugas server.
2. manajemen memori adalah suatu kegiatan untuk mengelola memori komputer. Proses ini menyediakan cara mengalokasikan memori untuk proses atas permintaan mereka, membebaskan untuk digunakan kembali ketika tidak lagi diperlukan serta menjaga alokasi ruang memori bagi proses.
3. Mereka dapat melindungi dari infeksi dari luar komputer atau jaringan yang dilindungi, dan membatasi aktivitas perangkat lunak berbahaya yang ada dengan memblokir permintaan masuk atau keluar pada port TCP/IP tertentu.

4. Terdapat 3 tipe penjadwal berada secara bersama-sama pada sistem operasi yang kompleks, yaitu:

Penjadwal jangka pendek (*short term scheduler*)

Bertugas menjadwalkan alokasi pemroses di antara proses-proses ready di memori utama. Penjadwalan dijalankan setiap terjadi pengalihan proses untuk memilih proses berikutnya yang harus dijalankan.

Penjadwal jangka menengah (*medium term scheduler*)

Setelah eksekusi selama suatu waktu, proses mungkin menunda sebuah eksekusi karena membuat permintaan layanan masukan/keluaran atau memanggil suatu *system call*. Proses-proses tertunda tidak dapat membuat suatu kemajuan menuju selesai sampai kondisi-kondisi yang menyebabkan tertunda dihilangkan. Agar ruang memori dapat bermanfaat, maka proses dipindah dari memori utama ke memori sekunder agar tersedia ruang untuk proses-proses lain. Kapasitas memori utama terbatas untuk sejumlah proses aktif. Aktivitas pemindahan proses yang tertunda dari memori utama ke memori sekunder disebut *swapping*. Proses-proses mempunyai kepentingan kecil saat itu sebagai proses yang tertunda. Tetapi, begitu kondisi yang membuatnya tertunda hilang dan dimasukkan kembali ke memori utama dan *ready*.

Penjadwal jangka panjang (*long term scheduler*)

Penjadwal ini bekerja terhadap antrian batch dan memilih batch berikutnya yang harus dieksekusi. *Batch* biasanya adalah proses-proses dengan penggunaan sumber daya yang intensif (yaitu waktu pemroses, memori, masukan/keluaran), program-program ini berprioritas rendah, digunakan sebagai pengisi (agar pemroses sibuk) selama periode aktivitas job-job interaktif rendah.

5. Manajemen file ini sangat penting bagi perusahaan karena sewaktu-waktu pasti dibutuhkan. Hal ini juga untuk memudahkan yang bersangkutan untuk menggunakannya. Filing ini juga sering digunakan sebagai ingatan seorang operator komputer atau manajer dalam kegiatan bisnis di sebuah perusahaan.

Tipe File pada Sistem Operasi

Secara umum ada tiga tipe file yang terdapat pada sistem operasi, di antaranya adalah:

1] Regular File

File reguler terdiri dari file teks dan biner. File teks ini berisi barisan teks dalam format txt. Sedangkan file biner berisi eksekusi (exe) dan juga biner dari berbagai program aplikasi dalam komputer.

File biner berbentuk exe hanya diketahui oleh sistem operasi. Sedangkan biner hasil program aplikasi hanya diketahui oleh program aplikasi yang menggunakan file tersebut.

2] Directory File

File direktori adalah file yang dimiliki oleh Operation System sebuah komputer. File ini berisi semua informasi terkait daftar file yang berada di dalam folder atau direktori.

3] Special File

File spesial adalah nama logik perangkat masukan (input) atau keluaran (output) yang dianggap sebagai file.

Contoh Kasus:

1. Anda bekerja sebagai administrator jaringan di sebuah perusahaan. Seorang pengguna melaporkan bahwa komputernya terinfeksi malware. Bagaimana Anda akan menggunakan perangkat lunak utility untuk mengidentifikasi dan menghilangkan malware dari komputer pengguna tersebut?
2. Sebagai administrator sistem, Anda mendapati bahwa ruang penyimpanan pada server Anda semakin berkurang. Bagaimana Anda akan menggunakan utilitas pembersihan disk untuk mengidentifikasi dan menghapus file sementara yang tidak diperlukan?
3. Seorang pengguna mengalami masalah dengan komputernya yang tampaknya terkait dengan driver perangkat keras. Bagaimana Anda akan menggunakan utilitas manajemen driver untuk mengidentifikasi masalah dan memperbarui driver yang sesuai?

Jawab:

1. Untuk mengidentifikasi dan menghilangkan malware dari komputer pengguna, berikut langkah-langkah yang dapat Anda lakukan sebagai administrator jaringan:
 - » Isolasi Komputer: Pertama, isolasi komputer pengguna dari jaringan Anda untuk mencegah penyebaran malware ke perangkat lain.
 - » Perangkat Lunak Antivirus: Gunakan perangkat lunak antivirus terbaru dan andalkan pada pemindaian menyeluruh untuk mengidentifikasi dan menghapus malware. Pastikan perangkat lunak antivirus Anda memiliki definisi virus yang terkini.
 - » Perangkat Lunak Anti-Malware: Selain antivirus, gunakan perangkat lunak anti- malware yang khusus dirancang untuk mendeteksi dan menghapus jenis malware yang lebih kompleks, seperti spyware dan adware.
 - » Pemindaian Sistem: Jalankan pemindaian lengkap pada komputer pengguna dengan perangkat lunak antivirus dan anti-malware. Biarkan perangkat lunak ini menemukan dan menghapus ancaman yang terdeteksi.
 - » Pemulihan Data: Jika ada data yang terpengaruh, pastikan untuk membackup data sebelum menghapus malware. Kemudian, pulihkan data dari backup setelah komputer bersih.
 - » Pemantauan Lanjutan: Setelah menghapus malware, pantau komputer untuk memastikan tidak ada aktivitas mencurigakan lainnya. Pastikan perangkat lunak keamanan selalu diperbarui.
 - » Edukasi Pengguna: Ajarkan kepada pengguna praktik keamanan yang baik, seperti tidak membuka lampiran email yang mencurigakan atau mengunduh file dari sumber yang tidak terpercaya.
 - » Tindakan Pencegahan: Evaluasi bagaimana komputer tersebut terinfeksi dan terapkan langkah-langkah pencegahan tambahan untuk mencegah serangan serupa di masa depan.
 - » Laporan Keamanan: Dokumentasikan insiden ini dan berikan laporan keamanan kepada manajemen dan tim keamanan jika diperlukan.

Penting untuk mencatat bahwa mengatasi infeksi malware dapat menjadi tugas yang rumit, tergantung pada tingkat keparahan infeksi. Jika Anda merasa kesulitan atau meragukan kemampuan Anda untuk mengatasi masalah ini, sebaiknya melibatkan tim keamanan TI yang lebih berpengalaman atau penyedia layanan keamanan yang kompeten.

2. Sebagai administrator sistem, untuk mengidentifikasi dan menghapus file sementara yang tidak diperlukan pada server Anda, Anda dapat menggunakan utilitas pembersihan disk seperti “Disk Cleanup” (untuk Windows) atau “du” dan “find” (untuk Linux). Di bawah ini, saya akan memberikan contoh kasus penggunaan “Disk Cleanup” pada Windows dan penggunaan perintah “du” dan “find” pada Linux:

Contoh 1:

Menggunakan Disk Cleanup pada Windows Buka “Disk Cleanup”:

Pada Windows, Anda dapat menemukan “Disk Cleanup” dengan cara mencarinya melalui menu “Start” atau dengan menjalankan perintah “cleanmgr” melalui Command Prompt.

Pilih Drive yang Akan Dibersihkan:

Disk Cleanup akan menawarkan Anda untuk memilih drive yang ingin Anda bersihkan. Pilih drive yang memiliki ruang penyimpanan yang terbatas.

Pilih Jenis File yang Akan Dihapus:

Anda akan melihat daftar jenis file yang dapat dihapus, seperti file sementara, file log, cache, dan lain-lain. Pilih jenis file yang ingin Anda hapus.

Konfirmasi dan Hapus:

Setelah Anda memilih jenis file yang akan dihapus, klik “OK” atau “Delete Files” untuk memulai proses pembersihan. Disk Cleanup akan menghapus file-file yang tidak diperlukan.

3. Untuk mengidentifikasi masalah driver perangkat keras dan memperbarui driver yang sesuai, Anda dapat menggunakan utilitas manajemen driver. Di bawah ini, saya akan memberikan contoh kasus penggunaan utilitas manajemen driver di Windows:

Contoh Kasus: Masalah Driver pada Kartu Grafis di Windows

Pengguna melaporkan bahwa komputernya mengalami masalah tampilan yang buram dan performa yang rendah saat menjalankan aplikasi yang memerlukan grafis yang tinggi, seperti game. Ini mungkin terkait dengan driver kartu grafis yang tidak terbaru atau rusak.

Identifikasi Model Kartu Grafis:

Pertama, Anda perlu mengetahui model kartu grafis yang digunakan oleh pengguna. Ini dapat ditemukan dengan membuka “Device Manager.” Tekan “Windows + X” dan pilih “Device Manager.”

Cek Status Kartu Grafis:

Di “Device Manager,” cari kategori “Display adapters” dan klik tanda panah di depannya untuk melihat model kartu grafis. Catat modelnya.

Unduh Driver Terbaru:

Kunjungi situs web produsen kartu grafis (misalnya, NVIDIA, AMD, atau Intel) dan temukan driver terbaru untuk model kartu grafis tersebut. Unduh driver yang sesuai untuk versi Windows yang digunakan oleh pengguna.

Instalasi Driver:

Instal driver yang telah diunduh. Klik dua kali pada file instalasi dan ikuti panduan yang muncul. Pastikan untuk memilih opsi instalasi yang sesuai.

Pembaruan Driver yang Sudah Ada:

Jika pengguna memiliki driver lama yang sudah diinstal, Anda juga dapat mencoba memperbarui driver tersebut melalui “Device Manager”:

Kembali ke “Device Manager.”

Klik kanan pada model kartu grafis di bawah “Display adapters.”

Pilih “Update driver” dan pilih opsi untuk mencari pembaruan secara otomatis. Uji Coba Kinerja:

Setelah driver baru diinstal atau driver yang sudah ada diperbarui, minta pengguna untuk menjalankan aplikasi atau game yang sebelumnya memiliki masalah grafis. Periksa apakah masalah telah teratasi.

Monitor Kinerja:

Anda juga dapat menggunakan utilitas monitor kinerja seperti “Task Manager” untuk memantau penggunaan CPU dan GPU saat aplikasi berjalan. Ini dapat membantu dalam mengidentifikasi masalah kinerja yang terkait dengan perangkat keras.

Laporan Masalah Lanjutan:

Jika masalah masih ada setelah pembaruan driver, lanjutkan dengan menganalisis masalah lebih lanjut, seperti memeriksa suhu perangkat keras atau kemungkinan masalah perangkat keras fisik.

Penting untuk selalu memastikan bahwa driver perangkat keras terbaru telah diinstal untuk mendapatkan kinerja yang optimal dan meminimalkan masalah yang terkait dengan driver. Juga, pastikan driver yang diunduh berasal dari situs resmi produsen perangkat keras untuk menghindari masalah kompatibilitas atau keamanan.

Bab 5

Transportasi

5.1 Pendahuluan Pengertian, Penjelasan dan Contoh

5.1.1 Pendahuluan

Terdapat bermacam macam model jaringan (*network model*). Suatu jaringan adalah suatu sistem garis garis atau saluran saluran yang menghubungkan titik titik yang berlainan. Beberapa contoh jaringan adalah jaringan rel kereta api, sistem saluran pipa, jaringan jalan raya, dan jaringan penerbangan. Dalam semua jaringan ini terjadi arus dari titik titik sumber menuju beberapa titik tujuan. Misalnya, dalam suatu sistem saluran pipa dapat dikirim air, minyak, atau gas dari sumber menuju langganan yang meminta.

Banyak masalah jaringan dapat dirumuskan sebagai masalah LP dan solusinya dapat diperoleh dengan menggunakan metode simpleks. Masalah transportasi adalah salah satu contoh dari model jaringan yang memiliki ciri-ciri seperti itu.

5.1.2 Penjelasan

Persoalan transportasi 1

- Persoalan transportasi terpusat pada pemilihan rute dalam jaringan distribusi produk antara pusat industri dan distribusi gudang atau antara distribusi gudang regional dan distribusi pengeluaran lokal.
 - Pada umumnya, masalah transportasi berhubungan dengan distribusi suatu produk tunggal dari beberapa sumber, dengan penawaran terbatas, menuju beberapa tujuan, dengan permintaan tertentu, pada biaya transpor minimum. Karena ada satu macam barang, suatu tempat tujuan dapat memenuhi permintaannya dari satu atau lebih sumber.
-

Persoalan transportasi 2

- Persoalan transportasi merupakan persoalan linier khusus yang disebut persoalan aliran network.
- Asumsi dasar model transportasi adalah bahwa biaya transpor pada suatu rute tertentu proporsional dengan banyaknya unit yang dikirimkan.
- Tujuan dari model transportasi adalah merencanakan pengiriman dari sumber-sumber ke tujuan sedemikian rupa untuk meminimumkan total biaya transportasi, dengan kendala- kendala:
- Setiap permintaan tujuan terpenuhi
- Sumber tidak mungkin mengirim komoditas lebih besar dari kapasitasnya.

5.2 Definisi Model Transportasi menurut Para Ahli

Metode Transportasi merupakan suatu metode yang digunakan untuk mengatur distribusi dari sumber-sumber yang menyediakan produk yang sama ke tempat-tempat yang membutuhkan secara optimal dengan biaya yang termurah. Alokasi produk ini harus diatur sedemikian rupa karena terdapat perbedaan biaya-biaya alokasi dari satu sumber atau beberapa sumber ke tempat tujuan yang berbeda.

Model transportasi berkaitan dengan suatu situasi dimana suatu komoditas yang ingin di kirim dari sejumlah sources (sumber) menuju ke sejumlah destination (tujuan). Tujuan dari masalah tersebut adalah menentukan jumlah komoditas yang harus di kirim dari tiap-tiap sumber ke tiap-tiap tujuan sedemikian hingga biaya total pengiriman dapat di minimumkan, dan pada saat yang sama pembatas yang berupa keterbatasan pasokan dan kebutuhan permintaan tidak dilanggar. Model transportasi mengasumsikan bahwa biaya pengiriman komoditas pada rute tertentu adalah proporsional dengan banyaknya unit komoditas yang di kirimkan pada rute tersebut (Setiawan dalam Nur Hamin, 2007).

Taha dan Fitri (1996) mengemukakan bahwa model transportasi berusaha menentukan sebuah rencana transportasi sebuah barang dari sejumlah sumber ke sejumlah tujuan. Data dalam model ini mencakup:

1. Tingkat penawaran di setiap sumber dan jumlah permintaan di setiap tujuan.
2. Biaya transportasi per unit barang dari setiap sumber ke setiap tujuan.

Model transportasi adalah suatu gambaran yang dituangkan ke dalam bentuk model matematika dari sebuah kasus transportasi yang dapat membantu berpikir secara cepat dan sistematis mengenai kasus tersebut. Bentuk umum dari model transportasi dapat digambarkan dalam bentuk matriks transportasi.

Pada umumnya masalah transportasi berhubungan dengan distribusi suatu produk tunggal dari beberapa sumber, dengan penawaran terbatas, menuju beberapa tujuan, dengan permintaan tertentu, pada biaya transportasi minimum. Karena hanya ada satu macam barang, suatu tempat tujuan dengan memenuhi permintaannya dari satu atau lebih sumber.

Asumsi dasar model ini adalah bahwa Biaya transportasi pada suatu rute tertentu proporsional dengan banyaknya unit yang dikirimkan. Definisi unit yang dikirimkan sangat tergantung pada jenis produk yang diangkut, yang penting, satuan penawaran dan permintaan akan barang yang diangkut harus konsisten.

Sebuah model transportasi dapat dibayangkan seperti contoh berikut. Misalnya, suatu produk yang dihasilkan pada tiga pabrik (sumber) harus didistribusikan ke tiga gudang (tujuan). Setiap pabrik memiliki kapasitas produksi tertentu, dan setiap gudang memiliki jumlah permintaan tertentu terhadap produk itu. Dengan diketahuinya biaya transportasi per unit dari masing-masing pabrik ke masing-masing gudang dengan tujuan meminimumkan biaya transportasi.

5.3 Manfaat dan Tujuan Transportasi

5.3.1 Tujuan Model Transportasi

Adapun tujuan dari penggunaan metode transportasi sebagai upaya untuk pendistribusian barang adalah agar supply dari beberapa sumber dapat dilakukan secara efektif dan biaya murah ke beberapa tempat tujuan. Sehubungan dengan hal tersebut, permasalahan yang akan dikaji selanjutnya adalah penggunaan metode transportasi yang dalam kajian ini membahas metode North West Corner sebagai salah satu teknik solusi dalam metode transportasi sebagai upaya untuk pendistribusian barang.

5.3.2 Manfaat Model Transportasi

1. Biaya distribusi dapat ditekan seminimal mungkin
2. Berguna untuk memecahkan permasalahan distribusi (alokasi hasil produksi)
3. Memecahkan permasalahan bisnis lainnya, yang berkaitan dengan pengiklanan, alokasi dana untuk investasi dsb.

5.4 Penerapan Model Transportasi

Terdapat beberapa cara dalam model transportasi atau metode distribusi, yaitu:

1. Untuk menentukan solusi awal dapat digunakan:
 - a| Metode North West Corner Metode Sudut Barat Laut
 - b| Metode Least Cost Metode Biaya Terkecil
 - c| Metode VAM Vogel's Approximation Method
2. Untuk menentukan solusi akhir yang yang optimal dapat diguakan:
 - a| Metode Modified Distribution MODI
 - b| Metode Stepping Stone SSM

5.5 Asumsi Model Masalah Transportasi

5.5.1 Asumsi Dasar

Model transportasi pada dasarnya merupakan sebuah program linier yang dapat dipecahkan oleh metode simpleks yang biasa. Tetapi strukturnya yang khusus memungkinkan pengembangan sebuah prosedur pemecahan, yang disebut teknik transportasi yang lebih efisien dalam hal perhitungan. Asumsi dasar dari model ini adalah bahwa biaya transportasi di sebuah rute tertentu adalah proposional secara langsung dengan jumlah unit yang dikirimkan. Definisi unit transportasi akan bervariasi bergantung pada jenis barang yang dikirimkan.

Model umum suatu persoalan transportasi dilandasi pada asumsi-asumsi berikut:

1. Bahwa suatu produk yang ingin diangkut tersedia dalam jumlah yang tetap dan diketahui
2. Bahwa produk tersebut akan dikirim melalui jaringan transportasi yang ada dengan memakai cara pengangkutan tertentu dari pusat-pusat permintaan.
3. Bahwa jumlah permintaan di pusat permintaan pun diketahui dalam jumlah tertentu dan tetap.
4. Bahwa ongkos angkutan per-unit produk yang diangkut pun diketahui, sehingga tujuan untuk meminimumkan biaya total angkutan dapat tercapai.

Karena hanya ada satu jenis komoditas, pada dasarnya setiap daerah tujuan dapat menerima komoditas dari sembarang daerah sumber.

5.6 Formulasi Model Masalah Transportasi

5.6.1 Formulasi Model Masalah Transportasi

Kendala model menunjukkan jumlah yang dapat ditawarkan oleh masing masing pabrik dan jumlah yang diminta pada setiap pasar sebagai jumlah dari alternatif. Alternatif pengiriman secara individu (*route*). Kendalanya berupa persamaan karena masalahnya seimbang (semua barang yang ditawarkan akan didistribusikan dan semua permintaan akan dipenuhi). Fungsi tujuan menunjukkan biaya total dengan C_{ij} sebagai biaya transfer dari pabrik i ke tujuan j , sehingga M_{us} umum masalah transportasi biaya termurah dengan m daerah asal dan n tujuan adalah:

Minimumkan Biaya Transpor Total:

$$\sum_{i=1}^m \sum_{j=1}^n C_{ij} X_{ij}$$

Dengan syarat $\sum_{i=1}^n X_{ij} = S_i$ (penawaran $i = 1, 2, \dots, m$)

$$\sum_{i=1}^m X_{ij} = D_j \text{ (permintaan } i = 1, 2, \dots, n)$$

semua $x \rightarrow 0$

Kondisi keseimbangan masalah transportasi diekspresikan sebagai

$$\sum_{i=1}^m S_i = \sum_{i=1}^m X_{ij} = \sum_{i=1}^n X_{ij} = \sum_{i=1}^m D_j$$

Masalah ini dapat dipecahkan dengan menggunakan metode Simpleks. Namun, suatu masalah transportasi yang relatif kecil, seperti pada contoh di atas, dapat berkembang menjadi tabel simpleks yang besar dengan 6 kendala, 9 variabel keputusan dan 6 artificial variabel. Jika diperhatikan lebih lanjut, semua koefisien matriks kendala pada masalah ini sama dengan 1. Sifat-sifat model ini mengakibatkan penggunaan metode solusi khusus yang perhitungannya lebih efisien dibanding metode Simplex.

5.7 Solusi Layak Basis Awal

Dari masalah yang telah disajikan, dapat diselesaikan melalui satu atau beberapa teknik solusi transportasi. Namun, untuk memulai proses solusi, suatu solusi dasar layak harus ditentukan.

Pada bentuk umum masalah transportasi di atas, terdapat m kendala penawaran dan n kendala permintaan, keseluruhannya terdapat $m + n$ kendala. Dalam suatu masalah LP banyaknya variabel basis dalam tabel simpleks sama dengan banyaknya kendala. Namun, pada masalah transportasi, terdapat sebuah kendala yang berlebih (*redundant*). Kondisi keseimbangan memberikan kenyataan bahwa jika $m + n - 1$ kendala terpenuhi kemudian $m + n$ persamaan juga akan terpenuhi. Hanya terdapat $m + n - 1$ persamaan independen. Sehingga, solusi awal hanya memiliki $m + n - 1$ variabel basis.

Ada beberapa metode untuk mencari solusi layak dasar awal. Tiga dari metode yang dikenal akan dibicarakan di sini, yaitu Metode VAM (Vogel Approximate Method) dan Metode MVAM (Modified Vogel's Approximate Method).

5.8 Metode VAM (*Vogel Approximate Method*)

VAM selalu memberikan suatu solusi awal yang lebih baik dibanding metode North west Corner dan sering kali lebih baik daripada metode Least-Cost. Kenyataannya, pada beberapa kasus, solusi awal yang diperoleh melalui VAM akan menjadi optimum. VAM melakukan alokasi dalam suatu cara yang akan meminimumkan penalty (opportunity cost) dalam memilih kotak yang salah untuk suatu alokasi. Proses VAM dapat diringkas sebagai berikut:

1. Hitung *opportunity cost* untuk setiap baris dan kolom. Opportunity cost untuk setiap baris i dihitung dengan mengurangkan nilai, terkecil pada baris itu dari nilai satu tingkat lebih besar pada baris yang sama. Opportunity cost kolom diperoleh dengan cara yang serupa. Biaya-biaya ini adalah penalty karena tidak memilih kotak dengan biaya minimum.
2. Pilih baris atau kolom dengan *opportunity cost* terbesar (jika terdapat nilai kembar, pilih secara sembarang). Alokasikan sebanyak mungkin ke kotak dengan nilai, minimum pada baris atau kolom yang dipilih. Untuk, terkecil, = minimum [,). Artinya penalti terbesar dihindari.
3. Sesuaikan penawaran dan permintaan untuk menunjukkan alokasi yang sudah dilakukan. Hilangkan semua baris dan kolom di mana penawaran dan permintaan telah dihabiskan.
4. Jika semua penawaran dan permintaan belum dipenuhi, kembali ke langkah 1 dan hitung lagi *opportunity cost* yang baru. Jika semua penawaran dan permintaan, solusi awal telah diperoleh.

Penerapan langkah-langkah ini pada contoh transportasi pupuk memberikan suatu alokasi VAM awal seperti ditunjukkan Tabel 5.1.

Tabel 5.1

Dari \ Ke	1	2	3	Supply
1	8	5	6	120
2	15	10	12	80
3	80 3	9	10	80
Demand	150	70	60	280

Penalty
Cost
Kolom

5 4 4

Sebagai suatu contoh perhitungan *penalty cost*, pikirkan baris pertama. Nilai, terkecil adalah 5 untuk. Kemudian yang satu tingkat lebih besar adalah, = 6. Sehingga *penalty cost* adalah beda antara dua nilai ini, $6 - 5 = 1$. Semua baris dan kolom yang lain dihitung dengan cara serupa.

Penalty cost terbesar untuk tabel ini adalah 6 yang terdapat pada baris 3. Alokasi pada baris ini dibuat pada kotak dengan nilai terkecil, dalam hal ini . Jumlah yang dialokasikan pada = minimum $[80, 150] = 80$. Sekarang tabel harus disesuaikan untuk menunjukkan sumber ke-3 telah terpakai habis dengan cara menghapus baris 3. Di samping itu, permintaan yang belum terpenuhi pada tujuan 1 menjadi 70 bukan lagi 150. Tabel yang disesuaikan dengan perhitungan ulang *penalty cost* dan alokasi kedua ditunjukkan pada Tabel 5.2.

Tabel 5.2

Ke Dari	1	2	3	Supply	Baris Penalty Cost
	1	8	5	6	
2	15	10	12	80	2
3	80	9	10	80	
Demand	150	70	60	280	
Penalty Cost Kolom		5	4	4	

Kolom 1 dipilih untuk alokasi kedua karena ia memiliki *penalty cost* baru (yang direvisi) terbesar, yaitu 7. Alokasi pada kolom ini dibuat pada kotak , karena ia memiliki nilai terkecil yaitu 8. Jumlah yang dialokasikan ke , = minimum $(70, 120) = 70$. Alokasi ini akan menghilangkan kolom 1 dan mengurangi penawaran baris 1 menjadi 50. Proses alokasi dan perhitungan kembali *penalty cost* diteruskan sampai semua penawaran dan permintaan terpenuhi. Solusi contoh transportasi pupuk yang diperoleh dengan VAM ditunjukkan pada Tabel 5.3.

Tabel 5.3

Ke Dari	1	2	3	Supply	Baris Penalty Cost		
1	70	8	5	50	6	120	1
2		15	10	10	12	80	2
3	80	3	9		10	80	
Demand	150	70	60	280			

Biaya Transport solusi ini adalah:

$$Z = (80 \times 70) + (6 \times 50) + (10 \times 70) + (12 \times 10) + (3 \times 80) = 1920$$

Biaya total untuk solusi awal 1920, adalah biaya awal terkecil yang diperoleh dari ketiga metode solusi awal. Kenyataannya solusi ini juga optimum, suatu keadaan yang akan ditunjukkan pada pembahasan mencari solusi optimum. Pada umumnya, VAM mengurangi banyaknya iterasi yang diperlukan untuk mencapai solusi optimum karena ia biasanya memberikan suatu solusi awal yang lebih baik daripada kedua metode yang lain.

5.9 Metode MVAM (*Modified Vogel's Approximate Method*)

Ada beberapa metode untuk menyelesaikan permasalahan transportasi, namun seiring dengan berkembangnya ilmu pengetahuan dan teknologi terdapat metode baru dalam menyelesaikan permasalahan transportasi, metode tersebut adalah Modified Vogel's Approximation Method (MVAM). Modified Vogel's Approximation Method (MVAM) merupakan salah satu hasil modifikasi dari VAM, MVAM menghasilkan solusi fisibel awal yang lebih optimal dari metode lain (Ullah, 2016).

Beberapa langkah yang dimodifikasi dari VAM, pertama mencari matriks tereduksi, selanjutnya mencari penalti dengan cara mencari selisih antara dua nilai terbesar pada setiap baris dan kolom, selanjutnya pilih penalti terbesar dan memilih sel dengan biaya transportasi matriks tereduksi) tertinggi. Menurut Ullah (2016) langkah-langkah Modified Vogel's Approximation Method sebagai berikut:

1. Mengurangi biaya transportasi terbesar pada setiap baris dengan masing-masing elemen setiap baris tabel transportasi dan letakkan di sebelah kiri atas elemen yang sesuai.
2. Mengurangi biaya transportasi terbesar pada setiap kolom dengan masing-masing elemen setiap kolom tabel transportasi dan letakkan disebelah kiri bawah elemen yang sesuai.
3. Membentuk matriks tereduksi yang elemennya adalah penjumlahan dari elemen kiri atas dan kiri bawah pada langkah satu dan langkah dua. Lalu tukar biaya transportasi yang awal dengan matriks tereduksi.
4. Menghitung penalti dengan mencari selisih dua angka terbesar pada setiap baris dan kolom.
5. Memilih nilai penalti tertinggi, jika ada dua atau lebih yang sama, pilih sel dengan biaya transportasi (matriks tereduksi) tertinggi.
6. Mengulangi langkah 4 dan 5 hingga semua alokasi terpenuhi.
7. Memasukkan semua alokasi sel dari matriks tereduksi ke tabel transportasi, lalu hitung hasil solusi fisibel awal.

Banyak penelitian yang membandingkan metode transportasi Modified Vogel's Approximation Method (MVAM) dengan North West Corner Method (NWCM), Least Cost Method (LCM), dan Vogel's Approximation Method (VAM). Analisisnya menghasilkan bahwa MVAM memberikan solusi awal yang lebih optimal dibandingkan dengan NCM, LCM, dan VAM. Lenti Amelia(2018) mengkaji metode transportasi Modified Vogel's Approximation Method (MVAM) dan Improved Zero Point Method (IZPM) untuk permasalahan setimbang dan tidak setimbang. Pada penelitiannya diperoleh hasil bahwa metode MVAM menghasilkan solusi yang lebih optimal dibandingkan dengan metode IZPM, baik pada saat data seimbang maupun data tidak seimbang.

5.10 Solusi Optimal Solusi Optimal

Setelah tabel awal transportasi dibuat (dengan sembarang metode). langkah berikutnya adalah mengecek apakah tabel tersebut sudah optimal. Menentukan entering dan leaving variable adalah tahap berikutnya dari pemecahan persoalan transportasi, setelah solusi fisibel awal diperoleh. Ada dua cara yang dapat

digunakan dalam menentukan entering dan leaving variable yaitu dengan menggunakan metode *Stepping Stone* dan *Modified Distribution Method* (Metode MODI).

5.10.1 Metode *Stepping Stone*

Metode *stepping stone* adalah suatu teknik yang berulang untuk berpindah dari suatu solusi awal yang layak ke solusi yang optimal dalam metode transportasi.

Syarat penyusunan jalur *stepping stone*:

1. Arah yang diambil, bisa searah jarum jam maupun berlawanan arah dalam membuat jalur tertutup.
2. Hanya ada satu jalur tertutup untuk setiap kotak kosong
3. Jalur harus hanya mengikut kotak terisi (dimana terjadi perubahan arah), kecuali pada kotak kosong yang sedang di evaluasi
4. Baik kotak terisi maupun kotak kosong dapat dilewati dalam penyusunan jalur tertutup.
5. Suatu jalur dapat melintas dirinya sendiri

Langkah-langkahnya:

- a) Memilih salah satu sel kosong (yang tidak mendapatkan alokasi).
 - b) Mulai dari sel ini, kita membuat jalur tertutup melalui sel-sel yang mendapatkan alokasi menuju sel kosong terpilih kembali. Jalur tertutup ini bergerak secara horizontal dan vertikal saja.
 - c) Mulai dengan tanda (+) pada sel kosong terpilih, kita menempatkan tanda (-) dan (+) secara bergantian pada setiap sudut jalur tertutup.
 - d) Menghitung indeks perbaikan dengan cara menjumlahkan biaya transportasi pada sel bertanda (+) dan mengurangi biaya transportasi pada sel bertanda (-).
5. Mengulangi tahap 1 sampai 4 hingga indeks perbaikan untuk semua sel kosong telah dihitung. Jika indeks perbaikan dari sel-sel kosong lebih besar atau sama dengan nol, solusi optimal telah tercapai.

Sebuah penambahan dan pengurangan yang sama besar harus kelihatan pada setiap baris dan kolom pada jalur itu.

5.10.2 Metode MODI (*Modified Distribution Method*)

Metode MODI sangat mirip dengan *Stepping Stone* kecuali bahwa MODI lebih efisien dalam menghitung indeks perbaikan sel kosong. Perbedaan utama antara dua metode ini yaitu pada langkah pemakaian jalur terpendek. Untuk menghitung perbaikan, metode *Stepping Stone* harus mencari jalur terpendek untuk tiap sel kosong.

Berikut RUMUS MODI:

$$R_i + K_j = C_{ij}$$

Di mana

R_i = Nilai baris ke- i

K_j = Nilai kolom ke- j

C_{ij} = Biaya pengangkutan 1 unit barang dari sumber i ke tujuan j

Kelebihan metode MODI adalah jumlah iterasi yang lebih sedikit dan indeks perbaikan dapat dicari tanpa harus mencari loop dari tiap-tiap sel kosong, hanya membutuhkan satu loop yang didapat setelah menentukan sel dengan indeks perbaikan terbesar.

Berikut merupakan langkah-langkah penggunaan metode MODI dalam menyelesaikan masalah transportasi adalah sebagai berikut

1. Tentukan tabel awal yang *feasible* dengan menggunakan metode Northwest-Corner atau metode Least Cost.
2. Tambahkan variabel R_i dan K_j pada setiap baris dan kolom
3. Cari nilai R_i maupun K_j untuk setiap sel basis (jalur yang terpakai) menggunakan rumus $R_i + K_j - C_{ij}$ dengan mengasumsikan salah satu nilai R_i atau $K_j - 0$.
4. Hitung semua nilai sel non basis atau indeks tiap jalur yang tidak terpakai dengan menggunakan rumus $I(ij) = C_{ij} - R_i - K_j$
5. Tentukan sel yang akan masuk basis dengan memilih nilai sel non basis yang memiliki negatif terbesar
6. Tabel optimum tercapai apabila sel bukan basis semuanya memiliki nilai 20
7. Jika tabel belum optimum, ulangi kembali langkah 2 sehingga ditemukan tabel optimum

5.11 Soal dan Kasus

Berikut ini akan diberikan sebuah contoh. Suatu perusahaan mempunyai tiga pabrik di W, H, O. Dengan kapasitas produksi tiap bulan masing-masing 90 ton, 60 ton, dan 50 ton; dan mempunyai tiga gudang penjualan di A, B, C dengan kebutuhan tiap bulan masing-masing 50 ton, 110 ton, dan 40 ton. Biaya pengangkutan setiap ton produk dari pabrik W, H, O ke gudang A, B, C adalah sebagai berikut:

Dari	Biaya tiap ton (dalam ribuan Rp)		
	Ke gudang A	Ke gudang B	Ke Gudang C
Pabrik W	20	5	8
Pabrik H	15	20	10
Pabrik O	25	10	19

Tentukan alokasi hasil produksi dari pabrik-pabrik tersebut ke gudang-gudang penjualan dengan biaya pengangkutan terendah.

Tabel Transportasi

Dari \ Ke	Ke gudang A		Ke gudang B		Ke gudang C		Kapasitas Pabrik
	X_{ij}		X_{ij}		X_{ij}		
Pabrik W	X_{11}	20	X_{12}	5	X_{13}	6	90
Pabrik H	X_{21}	15	X_{22}	10	X_{23}	12	60
Pabrik O	X_{31}	3	X_{32}	9	X_{33}	10	50
Kebutuhan gudang		50		110		40	200

X_{ij} adalah banyaknya alokasi dari sumber (pabrik) i ke tujuan (gudang) j . Nilai X_{ij} inilah yang akan kita cari.

Prosedur Alokasi

Pedoman prosedur alokasi tahap pertama adalah pedoman sudut barat laut (North West Corner Rule) yaitu pengalokasian sejumlah maksimum produk mulai dari sudut kiri atas (X_{11}) dengan melihat kapasitas pabrik dan kebutuhan gudang.

Alokasi tahap pertama dengan metode sudut barat laut

Dari \ Ke	Ke gudang A	Ke gudang A	Ke gudang A	Kapasitas Pabrik
Pabrik W	50	40	8	90
Pabrik H	15	60	10	60
Pabrik O	25	10	40	50
Kebutuhan Gudang	50	110	40	200

Biaya Pengangkutan untuk alokasi tahap pertama sebesar = $50(20) + 40(5) + 60(20) + 10(10) + 40(19) = 3260$.

5.12 Soal dan Kasus 2

Metode northwest corner – metode pojok barat laut

Pabrik sosis dengan tiga lokasi produksi masing-masing memproduksi 150 pcs, 175 pcs, 275 pcs. Pabrik tersebut akan mengirimkan ketiga pasar dengan masing-masing permintaan 200 pcs, 100 pcs, 300 pcs. Kapasitas penawaran pabrik, permintaan pada ketiga pasar dan biaya transpor perunit sebagai berikut:

Pabrik	Pasar			Penawaran
	I	II	III	
A	6	8	10	150
B	7	11	11	175
C	4	5	12	275
Permintaan	200	100	300	600

Jawaban:

	I	II	III	Penawaran
A	150 6	X 8	X 10	150
B	50 7	100 11	25 11	175
C	X 4	X 9	275 12	275
Permintaan	200	100	300	600

Solusi layak basis

$$\begin{aligned}
 Z &= 150(6) + 50(7) + 100(11) + 25(11) + 275(12) \\
 &= 900 + 350 + 1100 + 275 + 3300 \\
 &= 5925
 \end{aligned}$$

Bab 6

Theory Permainan

6.1 Pendahuluan

Teori permainan (game theory) adalah suatu pendekatan matematis untuk merumuskan situasi persaingan dan konflik antara berbagai kepentingan. Teori ini dikembangkan untuk menganalisis proses pengambilan keputusan dari situasi-situasi persaingan yang berbeda-beda dan melibatkan dua atau lebih kepentingan. Sebagai contoh para manajer pemasaran bersaing dalam memperebutkan bagian pasar, para pimpinan serikat dan manajemen yang terlibat dalam penawaran kolektif, para jendral tentara yang ditugaskan dalam perencanaan dan pelaksanaan perang, dan para pemain catur, yang semuanya terlibat dalam usaha untuk memenangkan permainan. Kepentingan-kepentingan yang bersaing dalam permainan disebut para pemain (players). Anggapannya adalah bahwa setiap pemain mempunyai kemampuan untuk mengambil keputusan secara bebas dan rasional.

Teori permainan mula-mula dikembangkan oleh seorang ahli matematika perancis bernama Emile Borel pada tahun 1921. Kemudian, Jhon Von Neumann dan Oskar Morgenstern mengembangkan lebih lanjut sebagai alat untuk merumuskan perilaku ekonomi yang bersaing. Aplikasi-aplikasi nyata yang paling sukses dari teori permainan banyak ditemukan dalam militer. Tetapi dengan berkembangnya dunia usaha (bisnis) yang semakin bersaing dan terbatasnya sumber daya serta saling ketergantungan social, ekonomi, dan ekologi yang semakin besar, akan meningkatkan pentingnya aplikasi-aplikasi teori permainan. Kontrak dan program tawar menawar serta keputusan-keputusan penetapan harga adalah contoh penggunaan teori permainan yang semakin meluas.

Model-model teori permainan dapat diklasifikasikan dengan sejumlah cara, seperti jumlah pemain, jumlah keuntungan dan kerugian dan jumlah strategi yang digunakan dalam permainan. Sebagai contoh, bila jumlah pemain adalah dua, permainan disebut sebagai permainan dua-pemain. Begitu juga, bila jumlah pemain adalah N (dengan $N \geq 3$), permainan disebut permainan N -pemain.

Bila jumlah keuntungan dan kerugian adalah nol, disebut permainan jumlah-nol atau jumlah-konstan. Sebaliknya, bila tidak sama dengan nol, permainan disebut permainan bukan jumlah-nol (non zero-sum game).

Teori permainan digunakan untuk mengambil keputusan pada situasi konflik dimana terdapat satu atau lebih pemain (lawan). Lawan atau pemain memiliki intelegensia yang sama. Setiap pemain mempunyai beberapa strategi untuk saling mengalahkan. Teori yang terkenal dari strategi ini adalah Two-Person Zero-Sum Game yaitu permainan dengan dua pemain dengan perolehan kemenangan (keuntungan) bagi salah satu pemain merupakan kehilangan (kerugian) bagi pemain lainnya. Jika jumlah kerugian dan keuntungan dari permainannya adalah nol, disebut sebagai permainan sejumlah nol (zero-sum game) atau permainan berjumlah konstan sebaliknya disebut sebagai permainan berjumlah bukan nol (non-zero sum game). Tidak setiap keadaan persaingan dapat disebut sebagai permainan (game). Kriteria atau ciri-ciri dari suatu permainan adalah:

1. Terdapat persaingan kepentingan di antara pemain,
2. Setiap pemain mempunyai sejumlah pilihan, terbatas atau tidak terbatas, yang disebut strategi,
3. Aturan permainan untuk mengatur pilihan-pilihan itu disebutkan satu-satu dan diketahui oleh semua pemain,
4. Hasil permainan dipengaruhi oleh pilihan-pilihan yang dibuat oleh semua pemain dan hasil untuk seluruh kombinasi pilihan dari pemain diketahui dan didefinisikan secara numerik.

Jadi, permainan (game) adalah suatu bentuk persaingan antara dua orang atau pihak atau antara dua kelompok atau grup yang saling berhadapan dan menggunakan aturan-aturan yang diketahui oleh kedua belah pihak yang saling berhadapan. Dalam permainan, pihak pertama disebut dengan pemain baris sedangkan pihak kedua disebut pemain kolom. Anggapannya adalah bahwa setiap pemain (individual atau kelompok) mempunyai kemampuan untuk mengambil keputusan secara bebas dan rasional. Aturan-aturan dalam permainan meliputi:

1. Langkah atau strategi yang dapat dipilih oleh tiap-tiap pemain,
2. Informasi yang digunakan oleh setiap pemain yang memilih langkah atau strategi,
3. Pembayaran, yang didefinisikan secara numerik, yang harus dipenuhi oleh setiap pemain setelah permainan selesai

Unsur-unsur dasar teori permainan sebagai berikut:

1. Angka-angka dalam matriks pay off (matriks permainan), menunjukkan hasil-hasil (pay offs) dari strategi-strategi permainan yang berbeda-beda.
2. Strategi permainan, adalah rangkaian kegiatan atau rencana yang menyeluruh dari seorang pemain sebagai reaksi atas aksi yang mungkin dilakukan oleh pemain lain yang menjadi pesaingnya.
3. Aturan-aturan permainan, menggambarkan kerangka dengan mana para pemain memilih strategi mereka.
4. Nilai permainan, adalah hasil yang diperkirakan per permainan atau pay off rata-rata dari sepanjang rangkaian permainan dimana kedua pemain mengikuti atau mempergunakan strategi mereka yang paling baik atau optimal.
5. Suatu strategi dikatakan dominan bila setiap pay off dalam strategi adalah superior terhadap setiap pay off yang berhubungan dalam suatu strategi alternative.
6. Suatu strategi optimal adalah rangkaian kegiatan atau rencana yang menyeluruh yang menyebabkan seorang pemain dalam posisi yang menguntungkan tanpa memperhatikan kegiatan-kegiatan para pesaingnya.
7. Tujuan dari model permainan adalah mengidentifikasi strategi atau rencana optimal untuk setiap pemain.

Konsep-konsep teori permainan paling tidak sangat penting untuk beberapa hal berikut ini:

1. Mengembangkan suatu kerangka untuk analisis pengambilan keputusan dalam situasi- situasi persaingan (dan kadang-kadang kerja sama).
2. Menguraikan suatu metode kuantitatif yang sistematis yang memungkinkan para pemain yang terlibat persaingan untuk memilih strategi-strategi yang rasional dalam pencapaian tujuan mereka.
3. Memberikan gambaran dan penjelasan fenomena situasi-situasi persaingan atau konflik.

Teori permainan digunakan untuk mengambil keputusan pada situasi konflik dimana terdapat satu atau lebih pemain (lawan). Lawan atau pemain memiliki intelegensia yang sama. Setiap pemain mempunyai beberapa strategi untuk saling

mengalahkan. Teori yang terkenal dari strategi ini adalah Two-Person Zero-Sum Game yaitu permainan dengan dua pemain dengan perolehan kemenangan (keuntungan) bagi salah satu pemain merupakan kehilangan (kerugian) bagi pemain lainnya. Jika jumlah kerugian dan keuntungan dari permainannya adalah nol, disebut sebagai permainan sejumlah nol (*zero-sum game*) atau permainan berjumlah konstan sebaliknya disebut sebagai permainan berjumlah bukan nol (*non-zero sum game*).

Salah satu jenis permainan adalah Dilema Tahanan dengan prinsip sebagai berikut:

1. Polisi menangkap 2 tersangka sebuah kasus kriminal.
2. Mereka diinterogasi secara terpisah, dan tidak ada komunikasi di antara mereka.
3. Karena bukti-bukti belum cukup, maka polisi memberi mereka 2 pilihan: menyangkal atau mengakui keterlibatan mereka berdua.
4. Jika keduanya menyangkal, maka A dan B akan mendapat hukuman penjara 1 tahun.
5. Jika A menyangkal dan B mengaku, maka A akan diganjar 10 tahun penjara, dan B bebas.
6. Jika A mengaku dan B menyangkal, maka A bebas dan B mendapat hukuman 10 tahun.
7. Jika keduanya mengaku, masing-masing akan diganjar 8 tahun.

Disebut menjadi Dilema Tahanan Kooperatif

1. Jika Napi A dan Napi B dapat mengadakan komitmen yang mengikat, maka mereka akan memilih {menyangkal, menyangkal}, dengan ganjaran masing-masing 1 tahun penjara. Dalam hal ini, *outcome*-nya lebih baik dibandingkan tanpa komitmen (game non kooperatif)
2. Game tetap harus bersifat simultan (A dan B bertindak secara serentak) dan informasi tidak sempurna (Baik A dan B tidak mengetahui apa pilihan lawannya, sebelum keduanya menetapkan pilihannya) Sebab jika A mengetahui B “menyangkal”, maka A jelas akan “mengaku”, sehingga A bebas (namun B dipenjara 10 tahun).
3. Jadi, *outcome* dari suatu game dapat ditingkatkan jika para pemain saling kooperatif

6.2 Definisi Theory Permainan Menurut Beberapa Ahli

Menurut Dimiyati (1992), teori permainan (*game theory*) adalah bagian dari ilmu pengetahuan yang berkaitan dengan pembuatan keputusan pada saat ada dua pihak atau lebih berada dalam kondisi persaingan atau konflik. Pihak-pihak yang bersaing ini disumsikan bersifat rasional dan cerdas, artinya masing-masing pihak akan melakukan strategi tindakan yang rasional untuk memenangkan persaingan itu, dan masing-masing pihak juga mengetahui strategi pihak lawannya. Selanjutnya pihak ini disebut pemain.

Menurut Ayu (1996), game theory merupakan suatu pendekatan matematis untuk merumuskan situasi persaingan dan konflik antara berbagai kepentingan. Game theory melibatkan dua atau lebih pengambil keputusan atau yang disebut pemain. Setiap pemain dalam game theory mempunyai keinginan untuk menang.

Teori permainan merupakan sebuah teori yang bertujuan untuk membantu memahami situasi dimana pengambil keputusan berinteraksi (Osborne, J. M., 2000). Teori permainan juga didefinisikan sebagai analisis umum mengenai strategi interaksi. Teori permainan berfokus pada penentuan strategi optimal dimana setiap pengambil keputusan mengambil keputusan secara rasional dan berusaha saling membaca strategi lawan (James. et al., 2000). Myerson dalam Nicola et al. (2015) mendefinisikan teori permainan sebagai studi tentang model matematika dari konflik dan kooperasi pengambil keputusan yang rasional dan intelektual. Sehingga, teori permainan merupakan studi mengenai pengambilan keputusan yang bertujuan menentukan strategi optimal.

Teori permainan (*games theory*) merupakan salah satu solusi dalam merumuskan keadaan persaingan antara berbagai pihak dan berbagai kepentingan. Pendekatan dalam teori permainan akan memberikan suatu gambaran yang sistematis dari para pelaku persaingan atau kita sebut para pemain, dalam memaksimalkan usaha untuk mencapai tujuannya. Teori ini menjadi terkenal oleh Jon Von Neumann dalam karyanya yang berjudul “Theory and Practice of Games and Economic Behaviour”, yang dipublikasikan pada tahun 1944.

6.3 Manfaat Theory Permainan

Manfaat tersebut tidak lepas dari orang yang terlibat serta yang pertama kali mengemukakan game theory, yaitu seorang ahli matematika Perancis Emile Borel (1921). Kemudian dikembangkan oleh John V.N dan Oscar Mogenstern lebih lanjut sebagai alat untuk merumuskan perilaku ekonomi yang bersaing. Teori permainan dikembangkan untuk menganalisa proses pengambilan keputusan yaitu strategi optimum dari situasi-situasi persaingan yang berbeda-beda dan melibatkan dua atau lebih kepentingan. Berikut Manfaat dari Game Theory:

1. Mengembangkan suatu kerangka untuk pengambilan keputusan dalam suatu persaingan
2. Menguraikan metode kuantitatif yang sistematis bagi pemain yang terlibat dalam persaingan untuk memilih strategi yang tradisional dalam pencapaian tujuan
3. Memberi gambaran dan penjelasan fenomena situasi persaingan / konflik.
4. Membuat strategi negosiasi, ataupun strategi dalam persaingan bisnis yang diwarnai oleh terjadinya konflik.

6.4 Strategi Murni (*Pure Strategy Game*)

Pada pure strategy game, pemain yang akan memaksimalkan akan mengidentifikasi strategi yang optimumnya dengan menggunakan kriteria maksimin, sedangkan pemain yang akan meminimumkan akan mengidentifikasi strategi optimumnya dengan menggunakan kriteria minimaks, maka permainan telah terpecahkan. (untuk menguji hal ini, nilai tersebut harus merupakan nilai maksimum bagi kolom yang bersangkutan, dan sekaligus merupakan nilai minimum bagi baris yang bersangkutan). Dalam kasus seperti ini maka telah mencapai titik keseimbangan. Titik ini dikenal dengan titik sadel (saddle point).

Jika nilai maksimin tidak sama dengan nilai minimaks, maka titik keseimbangan tidak akan dapat tercapai. Hal ini berarti bahwa saddle point-nya tidak ada dan permainan tidak dapat diselesaikan dengan strategi murni.

Kriteria maksimin dan kriteria minimaks adalah sebagai berikut:

1. Kriteria maksimin (untuk pemain yang memaksimumkan)

Dapatkan nilai minimum dari masing-masing baris. Nilai terbesar (nilai maksimum) dari nilai-nilai minimum ini adalah nilai maksimin. Dengan demikian, maka untuk permainan dengan strategi murni ini, strategi optimumnya adalah baris tempat nilai maksimin tersebut.

2. Kriteria minimaks (untuk pemain yang meminimumkan)

Dapatkan nilai maksimum pada masing-masing kolom. Nilai terkecil (nilai minimum) dari nilai-nilai maksimum ini adalah nilai minimaks. Dengan demikian, maka untuk permainan dengan strategi murni ini, strategi optimumnya adalah kolom tempat nilai minimaks terletak.

Contoh 2.1

Dua buah perusahaan mempunyai strategi yang berbeda untuk menarik konsumen, perusahaan A mempunyai 2 buah strategi dan perusahaan B mempunyai 3 buah strategi. Struktur strategi dan pay-off-nya adalah sebagai berikut:

Tabel 6.1 Matriks Permasalahan *Pure Strategy Game*

	Perusahaan B			
		B_1	B_2	B_3
Perusahaan A	A_1	3	4	4
	A_2	9	5	6

Berdasarkan kasus di atas, bagaimana strategi yang harus digunakan oleh masing-masing pemain atau perusahaan agar masing-masing mendapatkan hasil yang optimal?

Penyelesaian:

Langkah 1

Untuk pemain baris (perusahaan A), pilih nilai yang paling kecil untuk setiap baris (baris satu nilai terkecilnya 3, dan baris dua nilai terkecilnya 5). Selanjutnya dari dua nilai terkecil tersebut, pilih nilai yang paling besar, yaitu nilai 5.

Langkah 2

Untuk pemain kolom (perusahaan B), pilih nilai yang paling besar untuk setiap kolom (kolom satu nilai terbesarnya 9, kolom dua nilai terbesarnya 5, kolom tiga nilai terbesarnya 6). Selanjutnya dari tiga nilai terbesar tersebut, pilih nilai yang paling kecil, yaitu nilai 5.

Tabel 6.2 Matriks *Pay-Off* Strategi Murni (Maksimum dan Minimaks)

	Perusahaan B				
		B ₁	B ₂	B ₃	Minimum Baris
Perusahaan A	A ₁	3	4	4	3
	A ₂	9	5	6	5 (Maksimsin)
		9	5 (Minimaks)	6	

Langkah 3

Karena pemain baris A dan pemain kolom B sudah sama, yakni masing-masing memiliki nilai 5, maka permainan ini sudah dapat dikatakan optimal, sudah ditemukan nilai permainan (*saddle point*) yang sama.

6.5 Strategi Campuran (*Mixed Strategy Game*)

Teori permainan dengan jumlah nol dari dua pemain (zero sum games) adakalanya tidak mempunyai titik pelana pada matriks *pay-off*-nya, sehingga keseimbangan akan dicari dengan cara lain, yaitu dengan permainan strategi campuran. Setiap pemain seringkali tidak mengetahui strategi apa yang dipilih oleh pemain lawan, sehingga dia harus memutuskan suatu strategi yang akan minimal berakibat sama dengan strategi yang dipilih oleh pemain lain. *Pay-off* yang akan coba didapat adalah sama caranya dengan pure strategy yaitu menggunakan konsep Maksimin untuk A (baris) dan konsep Minimaks untuk B (kolom). Dalam suatu permainan yang diselesaikan dengan strategi campuran, strategi dari setiap pemain akan mempunyai probabilitas yang menunjukkan proposi waktu atau banyaknya bagian yang dipergunakan untuk melakukan strategi tersebut. Pemilihan strategi akan dilakukan secara acak dari beberapa pilihan strategi yang ada. Setiap strategi yang akan dipilih akan ditentukan peluang berupa persentase dari tiap strategi yang dipilih. Peluang ini penting digunakan sebagai pedoman akan prioritas strategi yang akan dilakukan. Peluang yang ditentukan bisa merupakan

pengalaman dari pengambil keputusan akan keputusan-keputusan yang pernah dilakukan oleh lawan, atau berdasarkan penelitian yang dilakukan akan kejadian masa depan dari suatu keputusan. Misalkan,

P : peluang pemain I menggunakan strategi S_{11} . Maka,

$1 - P$: peluang pemain I tidak menggunakan strategi S_{11} . Sebaliknya jika,

Q : peluang pemain II menggunakan strategi S_{21} . Maka,

$1 - Q$: peluang pemain II tidak menggunakan strategi S_{21} .

6.6 Aturan Dominansi pada Theory Permainan

Sebelum menyelesaikan suatu permainan perlu dipertimbangkan apakah ada baris atau kolom dalam matriks pembayarannya yang tidak efektif pengaruhnya di dalam penentuan strategi optimum dan nilai permainan. Bila ada maka baris atau kolom yang seperti itu bisa dihapus atau tidak dipakai, Hal itu berarti bahwa probabilitas untuk memilih strategi sesuai baris atau kolom tersebut sama dengan nol. Dengan demikian ukuran matriks pembayaran yang tersisa akan lebih kecil. Hal ini akan lebih mempermudah untuk menyelesaikannya. Aturan demikian ini dinamakan aturan dominansi.

- a) Aturan dominansi bagi pemain pertama P1 (pemain baris). Karena pemain P1 (pemain baris) merupakan pemain yang berusaha untuk memaksimalkan kemenangan/perolehannya maka aturan dominansinya adalah sebagai berikut: bila terdapat suatu baris dengan semua elemen dari baris tersebut adalah sama (sekolom) dari baris yang lain maka baris tersebut dikatakan didominasi dan baris itu telah dihapus.
- b) Aturan dominansi bagi pemain kedua P2 (pemain kolom). Karena pemain kedua P2 merupakan pemain yang berusaha untuk meminimumkan kekalahan/kerugiannya maka aturan dominansinya adalah sebagai berikut: bila terdapat suatu kolom dengan semua elemen dari kolom tersebut adalah sama atau lebih besar dari elemen dalam posisi yang sama (sebaris) dari kolom yang lain maka kolom tersebut dikatakan didominasi dan kolom itu dapat dihapus. Aturan dominansi ini dapat diulang lagi jika masih ada baris atau kolomnya yang didominasi oleh baris atau kolom yang lain. Dan ini memungkinkan matriks pembayaran semula akan tersisa menjadi matriks pembayaran dengan satu elemen saja. Bila hal ini dapat terjadi

maka permainannya dapat diselesaikan dengan strategi murni dengan nilai permainan sesuai dengan elemen yang tersisa tersebut. Tetapi tidak semua permainan yang mempunyai titik pelana dapat diselesaikan dengan aturan dominansi yang berulang-ulang tersebut.

6.6.1 Strategi Terdominasi dan Strategi Dominan

Strategi terdominasi adalah strategi yang inferior terhadap sejumlah strategi lain, untuk setiap strategi yang dipilih lawan (dengan kata lain, pay-off strategi tersebut \leq pay-off sejumlah strategi lainnya).

Strategi dominan adalah strategi yang memiliki payoff tertinggi dibandingkan dengan strategi lainnya. Misalkan strategi "X" adalah strategi dominan bagi pemain A, maka apapun strategi yang dipilih pemain B, pemain A tetap akan memilih strategi "X". Strategi mendominasi adalah strategi X dikatakan mendominasi strategi Y jika payoff strategi $X \geq$ payoff strategi Y, untuk setiap strategi yang dipilih lawan.

6.7 Prosedur Penyelesaian Grafik pada Theory Permainan

Pandangan permainan dengan suatu strategi campuran, setelah strategi yang didominasi strategi lain dihilangkan, maka pemain tinggal memilih 2 strategi murni. Misalkan pemain tersebut adalah pemain I. karena strategi campuran adalah dan , maka perlu dicari penyelesaian optimal untuk , membuat plot harapan payoff sebagai fungsi dari untuk setiap strategi murni lawan sangat mudah. Grafik ini dapat digunakan untuk mengidentifikasi titik yang memaksimalkan harapan payoff minimum. Strategi campuran minimaks lawan juga dapat ditunjukkan oleh grafik tersebut. Untuk menggambarkan prosedur ini, perhatikan Variasi 3, masalah kampanye politik (lihat Tabel 6.2). Ingat kembali strategi murni ke 2 sehingga table payoff dapat disederhanakan menjadi tabel 12-6. Oleh sebab itu untuk setiap strategi murni yang berlaku untuk pemain II, harapan payoff untuk pemain I menjadi:

Tabel 6.3

(y_1, y_2, y_3)	Harapan <i>Payoff</i>
(1, 0, 0)	$0x_1 + 5(1 - x_1) = 5 - 5x_1$
(0, 1, 0)	$-2x_1 + 4(1 - x_1) = 4 - 6x_1$
(0, 0, 1)	$2x_1 - 3(1 - x_1) = -3 + 5x_1$

Sekarang, gambar garis harapan payoff ditunjukkan untuk setiap nilai diberikan dan harapan payoffkan sesuai dengan:

$$\text{Expected payoff for player 1} = y_1(5 - 5x_1) + y_2(4 - 6x_1) + y_3(-3 + 5x_1)$$

Minimum harapan payoff ditunjukkan oleh titik pada garis terbawah. Berdasarkan kriteria minimaks (atau maksimin), pemain I harus memilih nilai yang memberikan harapan payoff minimum yang terbesar sehingga

$$v = v = \max_{0 < x_1 < 1} \{ \min(-3 + 5x_1, 4 - 6x_1) \}$$

Jadi nilai optimal untuk x_1 adalah titik potong kedua garis $(-3 + 5x_1)$ dan $(4 - 6x_1)$. Selesaikan persamaan

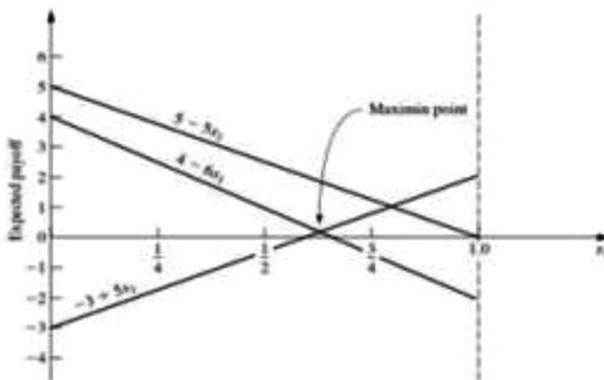
$$-3 + 5x_1 = 4 - 6x_1$$

Diperoleh $X_1 = \frac{7}{11}$ dan $(X_1, X_2) = (\frac{7}{11}, \frac{4}{11})$ adalah strategi campuran optimal untuk pemain I, serta dari permainan:

$$\underline{v} = -3 + 5\left(\frac{7}{11}\right) = \frac{2}{11}$$

Table payoff dipersingkat untuk variasi 3 dari masalah tersebut

			Player 2		
			y_1	y_2	y_3
Player 1	Probability	Pure Strategi	1	2	3
	x_1	1	0	-2	2
	$1 - x_1$	2	5	4	-3



Untuk mencari strategi optimal yang sesuai dengan pemain II, perhatikan cara berikut ini. Menurut definisi nilai minimaks dan teorema minimaks payoff yang dihasilkan dari strategi optimal $(y_1, y_2, y_3) = (y_1, y_2, y_3)$ akan memenuhi kondisi:

$$y_1^*(5 - 5x_1) + y_2^*(4 - 6x_1) + y_3^*(-3 - 5x_1) \leq \bar{v} = v = \frac{2}{11}$$

Untuk setiap nilai $x_1 (0 < x_1 < 1)$. Bila pemain I bermain optimal (yaitu $x_1 = \frac{2}{11}$), pertaksamaan ini menjadi persamaan, sehingga:

$$\frac{20}{11}y_1^* + \frac{2}{11}y_2^* + \frac{2}{11}y_3^* = v = \frac{2}{11}$$

Karena (y_1, y_2, y_3) adalah distribusi peluang maka

$$y_1^* + y_2^* + y_3^* = 1$$

Akibatnya, diperoleh $y_1 = 0$, karena jika $y_1 > 0$ akan bertentangan dengan 2 persamaan terakhir yaitu harapan payoff pada grafik di $x_1 = \frac{7}{11}$ ada di atas titik maksimin. (secara umum, sembarang garis yang tidak melalui titik maksimin harus diberikan bobot 0, untuk menghindari harapan payoff naik di atas titik tersebut).

Jadi

$$y_2^*(4 - 6x_1) + y_3^*(-3 + 5x_1) \begin{cases} \leq \frac{2}{11} & \text{untuk } 0 \leq x_1 \leq 1 \\ = \frac{2}{11} & \text{untuk } x_1 = \frac{7}{11} \end{cases}$$

Tetapi y_2^* dan y_3^* adalah bilangan, sehingga bagian kiri adalah persamaan garis lurus yang merupakan rata-rata tertimbang dengan bobot 2 garis dibawah dari grafik karena ordinat garis ini harus sama dengan $\frac{2}{11}$ di x_1 , dan karena ordinat tidak boleh lebih dari $\frac{2}{11}$, maa garis tersebut harus horizontal. (kesimpulan ini selalu benar, kecuali nilai optimal untuk x_1 adalah 0 atau 1, dalam hal ini pemain II juga harus menggunakan satu strategi murni). Sehingga didapat

$$y_2^*(4 - 6x_1) + y_3^*(-3 + 5x_1) = \frac{2}{11} \text{ untuk } 0 \leq x_1 \leq 1$$

Untuk menyelesaikan y_2^* dan y_3^* , pilih dua nilai (misalkan 0 dan 1) dan selesaikan persamaan yang diperoleh

$$\begin{aligned} 4y_2^* - 3y_3^* &= \frac{2}{11} \\ -2y_2^* + 3y_3^* &= \frac{2}{11} \end{aligned}$$

Penyelesaian $y_2^* = \frac{5}{11}$, dan $y_3^* = \frac{6}{11}$ jadi strategi campuran optimal untuk pemain II adalah $(y_1, y_2, y_3) = (0, \frac{5}{11}, \frac{6}{11})$.

Jika pada masalah lain terdapat lebih dari 2 garis melalui titik maksimin, maka terdapat lebih dari dua y_1^* yang lebih dari kondisi ini akan mengakibatkan terdapat banyak nilai yang sama untuk strategi campuran optimal bagi pemain II. Salah satu strategi seperti itu dapat diidentifikasi dengan membuat semua nilai 0 kecuali untuk ke dua y_1^* ini, kemudian selesaikan persamaan yang diperoleh. Untuk kedua persamaan tersebut, garis yang bersesuaian dengan harus mempunyai gradient positif dan garis lainnya bergradien negative.

6.8 Penyelesaian Theory Permainan dengan Program Linear

Setiap permainan dengan strategi campuran dapat diselesaikan dengan mengubah masalah menjadi pemrograman linear. Seperti akan kita lihat, transformasi ini membutuhkan sedikit lebih banyak jika dibandingkan dengan menerapkan teorema minimaks dan menggunakan definisi nilai maksimin \underline{v} dan nilai minimaks \bar{v} . Pertama, perhatikan bagaimana mencari strategi optimal untuk pemain 1.

$$\text{Expected payoff for player 1} = \sum_{i=1}^m \sum_{j=1}^n p_{i,j} x_i y_j$$

Dan strategi (y_1, y_2, \dots, y_m) optimal jika

$$= \sum_{i=1}^m \sum_{j=1}^n p_{i,j} x_i y_j \geq \underline{v} = v$$

Untuk setiap strategi lawan (y_1, y_2, \dots, y_n) . Jadi pertidaksamaan ini harus dipenuhi, sebagai contoh, untuk setiap strategi murni pemain II, yaitu untuk setiap strategi (y_1, y_2, \dots, y_n) dimana satu $y_j = 1$ dan sisanya 0. Substitusi nilai ke pertaksamaan, diperoleh

$$\sum_{i=1}^m p_{ij} x_i \geq v \text{ untuk } j = 1, 2, \dots, n$$

Jadi dari satu pertaksamaan diperoleh n pertaksamaan. Selanjutnya, n pertaksamaan mengakibatkan:

$$\sum_{j=1}^n y_j \left(\sum_{i=1}^m p_{ij} x_i \right) \geq \sum_{j=1}^n y_j v = v$$

Karena

$$\sum_{j=1}^n y_j = 1$$

Karena implikasi ini berlaku dua arah, maka menyelesaikan n pertaksamaan di atas ekuivalen dengan menyelesaikan pertaksamaan baru yang berlaku untuk setiap strategi (y_1, y_2, \dots, y_n) . Tetapi n pertaksamaan ini merupakan kendala pemrograman linear seperti kendala tambahan

$$\begin{aligned}x_1 + x_2 + \dots + x_n &= 1 \\x_i &\leq 0, \text{ untuk } i = 1, 2, \dots, m\end{aligned}$$

Yang dibutuhkan untuk menjamin bahwa x_1 adalah peluang. Jadi setiap penyelesaian (x_1, x_2, \dots, x_n) yang memenuhi kendala pemrograman linear adalah strategi optimal yang diinginkan. Konsekuensinya, masalah mencari strategi campuran optimal telah diubah menjadi mencari penyelesaian layak untuk masalah program linear yang dapat diselesaikan dengan cara yang telah dijelaskan dalam Bab 4. Tetapi masalah ada 2 hal, yaitu (1) v tidak diketahui dan (2) masalah program linear belum mempunyai fungsi tujuan. Dua hal ini dapat diselesaikan dengan menggantikan konstanta v yang tidak diketahui dengan perubahan x_{m+1} kemudian maksimumkan x_{m+1} sehingga x_{m+1} otomatis akan sama dengan v (menurut definisi) pada penyelesaian optimal untuk masalah pemrograman linear. Dapat disimpulkan bahwa, pemain I harus mencari strategi campuran optimal dengan menggunakan metode simplek. Kemudian memperoleh strategi campuran optimal pemain II melalui model permainan pertama, maka tidak perlu menyelesaikan model kedua secara langsung. secara umum kita selalu dapat mencari strategi campuran-campuran optimal untuk kedua pemain dengan memilih salah satu model, kemudian dengan menggunakan metode simplek diperoleh penyelesaian optimal dan penyelesaian optimal dual.

Rangkuman

1. Teori Permainan digunakan untuk mencari strategi terbaik dalam suatu aktivitas, dimana setiap pemain di dalamnya sama-sama mencapai utilitas tertinggi. Penerapannya banyak dilakukan di berbagai disiplin ilmu seperti biologi, militer, politik, diplomasi dan ilmu sosial.
2. Teori permainan pertama kali dikembangkan oleh ilmuwan Prancis bernama Emile Borel digunakan untuk menyelesaikan masalah yang berkaitan dengan tindakan sebuah unit bisnis.
3. Teori permainan dapat diselesaikan dengan strategi murni, strategi campuran dan juga dapat diselesaikan dengan program linear.

6.9 Soal dan Kasus

Soal 1

Pengusaha A dan B merebut pasar, mereka saling bersaing dengan menggunakan informasi pasar yang di peroleh riset pemasaran. A dapat memilih 4 daerah potensial dan B memilih 2 daerah potensial. Jika B memilih daerah 1 maka, keuntungan bagi A di daerah 1,2,3 berturut-turut adalah 3, 10, 3 sedangkan jika saat B memilih daerah 4 , maka A akan rugi 2, sedangkan jika B memilih daerah 2 maka keuntungan bagi A di daerah 1, 2, 3 dan 4 adalah sebanyak 4, 6, 2, 6. a) Buat tabel matrik A dan B, b) Strategi apa yang harus dilakukan oleh perusahaan A dan B.

Penyelesaian:

Dari Tabel dapat diselesaikan dengan strategi murni:

		Pengusaha B		Minimum	
		1	2		
Pengusaha A	Daerah Pemasaran				
	1	3	4	3	
	2	10	6	6	Maksimin
	3	3	2	2	
	4	-2	6	-2	
		Maksimum	10	6	Saddle Point
				Minimaks	

Strategi yang harus dilakukan atau yang harus dipilih oleh perusahaan A dan B untuk mendapatkan keuntungan yang maksimum adalah perusahaan A harus memilih daerah pemasaran 2, dan perusahaan B juga memilih daerah pemasaran 2.

Soal 2

Dua perusahaan konstruksi bersaing untuk mendapatkan keuntungan dari pangsa pasar yang ada, dengan mengandalkan strategi yang dimiliki. Putra karya mengandalkan 3 strategi dan permata karya menggunakan 4 strategi.

Penyelesaian:

		Perusahaan Permata Karya			
		Strategi harga murah (S1)	Strategi harga sedang (S2)	Strategi harga mahal (S3)	Strategi harga mahal (S4)
Perusahaan Putra Karya	Strategi harga murah (S1)	2	1	3	4
	Strategi harga mahal (S2)	4	10	13	5
	Strategi harga mahal (S3)	9	7	8	6

Langkah 1

Untuk pemain baris (Perusahaan PUTRA KARYA), pilih nilai paling kecil (baris 1 adalah 1, baris ke 2 adalah 4, dan baris ke 3 adalah 6). Selanjutnya dari dari ke tiga nilai terkecil tersebut, pilih angka yang paling besar yaitu 6.

		Perusahaan Permata Karya				Maximin
		Strategi harga murah (S1)	Strategi harga sedang (S2)	Strategi harga mahal (S3)	Strategi harga mahal (S4)	
Perusahaan Putra Karya	Strategi harga murah (S1)	2	1	3	4	> 1
	Strategi harga mahal (S2)	4	10	13	5	> 4
	Strategi harga mahal (S3)	9	7	8	6	> 6

Langkah 2

Untuk pemain kolom (Perusahaan PERMATA KARYA), pilih nilai paling besar (kolom 1 adalah 9, kolom 2 adalah 10, kolom 3 adalah 13, dan kolom 4 adalah 6). selanjutnya dari keempat nilai terbesar tersebut, pilih angka yang paling kecil yaitu 6 (rugi yang paling kecil).

		Perusahaan Permata Karya				Maximin
		Strategi harga murah (S1)	Strategi harga sedang (S2)	Strategi harga mahal (S3)	Strategi harga mahal (S4)	
Perusahaan Putra Karya	Strategi harga murah (S1)	2	1	3	4	> 1
	Strategi harga mahal (S2)	4	10	13	5	> 4
	Strategi harga mahal (S3)	9	7	8	6	> 6
Minimax	>	9	10	13	6	

Dapat disimpulkan:

- Pemain baris dan pemain kolom sudah memiliki pilihan strategi yang sama yaitu nilai 6 \Rightarrow optimal.
- Pilihan tersebut berarti bahwa meskipun perusahaan PUTRA KARYA menginginkan keuntungan yang lebih besar, tapi tetap hanya akan memperoleh keuntungan maksimal 6 dengan strategi harga mahal (S3), demikian juga dengan perusahaan PERMATA KARYA, kerugian yang paling minimal adalah 5, dengan merespon strategi perusahaan PUTRA KARYA, dengan strategi harga mahal (S4).
- Penggunaan strategi lain berdampak menurunkan keuntungan perusahaan PUTRA KARYA dan meningkatnya kerugian perusahaan PERMATA KARYA.

Soal 3

Dua perusahaan bersaing untuk mendapatkan keuntungan dari pangsa pasar yang ada, dengan mengandalkan strategi yang dimiliki. A mengandalkan 2 strategi dan B menggunakan 3 strategi.

		Perusahaan B		
		Strategi harga murah (S1)	Strategi harga sedang (S2)	Strategi harga mahal (S3)
Perusahaan A	Strategi harga murah (S1)	1	9	2
	Strategi harga mahal (S2)	8	5	4

Penyelesaian:

		Perusahaan B		
		Strategi harga murah (S1)	Strategi harga sedang (S2)	Strategi harga mahal (S3)
Perusahaan A	Strategi harga murah (S1)	1	9	2
	Strategi harga mahal (S2)	8	5	4

Langkah 1

Untuk pemain baris (Perusahaan A), pilih nilai yang paling kecil untuk setiap baris (baris satu nilai terkecilnya 1 dan baris dua nilai terkecilnya 4). Selanjutnya dari dua nilai terkecil tersebut, pilih nilai yang paling baik atau besar, yakni nilai 4.

Langkah 2

Untuk pemain kolom, (perusahaan B), pilih nilai yang paling besar untuk setiap kolom (kolom satu nilai terbesarnya 8, kolom dua nilai terbesarnya 9, dan kolom tiga nilai terbesarnya 4). Selanjutnya dari tiga nilai terbesar, pilih nilai yang kecil untuk perusahaan B, yakni 4.

		Perusahaan B			Maximin
		Strategi harga murah (S1)	Strategi harga sedang (S2)	Strategi harga mahal (S3)	
Perusahaan A	Strategi harga murah (S1)	1	9	2	→ 1
	Strategi harga mahal (S2)	8	5	4	→ 4
Minimax	→	8	9	4	

Dapat disimpulkan berdasarkan Tabel:

- Pemain baris dan pemain kolom sudah memiliki pilihan strategi yang sama yaitu nilai 4 Sehingga Nilai 4 Sebagai hasil Optimal.
- Pilihan tersebut berarti bahwa meskipun A menginginkan keuntungan yang lebih besar, tapi tetap hanya akan memperoleh keuntungan maksimal 4 dengan strategi harga mahal (S2), demikian juga dengan B, kerugian yang paling minimal adalah 4, dengan merespon strategi A, dengan strategi harga mahal (S3).
- Penggunaan strategi lain berdampak menurunnya keuntungan A dan meningkatnya kerugian B.

Bab 7

Teori Dualitas

7.1 Pendahuluan

Dari berbagai perkembangan matematika tersebut lahirlah berbagai situasi-situasi yang diharapkan akan menghasilkan keadaan yang optimal terhadap hasil yang diharapkan. Namun terkadang tidak lain suatu peristiwa akan menghasilkan dua kemungkinan hasil. Di mana kejadian awal atau mula-mula ini disebut dengan primal dan kejadian kedua disebut dengan dual. Maka berkembanglah teori primal-dual atau biasa disebut teori dualitas. Dimana teori dualitas ini sangat berguna bagi penyelesaian kasus-kasus program linear dengan model persamaan matematika.

Sedangkan Program linear sendiri adalah suatu metode penentuan nilai optimum dari suatu persoalan linear. Setiap permasalahan linear mempunyai permasalahan yang kedua yang berhubungan dengannya. Kedua permasalahan tersebut sangat dekat berhubungan, sehingga solusi optimal disatu permasalahan menghasilkan informasi yang lengkap untuk solusi optimal yang lainnya.

Dalam kenyataannya, sebuah perusahaan tidak selalu berhadapan dengan masalah- masalah yang dapat dipecahkan secara sederhana menggunakan LP. Berdasarkan permasalahan ini, dikembangkan teknik dualitas yaitu menjelaskan secara sistematis bahwa sebuah kasus LP terdiri dari masalah primal dan dual. Bentuk primal adalah bentuk awal dari suatu permasalahan program linier, sedangkan dualnya merupakan bentuk duplikatnya. Konsep primal dan dual dapat diartikan sebagai lawan dari, artinya jika ada bentuk primal dari suatu permasalahan program linier maka akan terdapat bentuk lawannya yaitu dual.

Tabel simpleks mengandung informasi ekonomi tambahan yang bahkan lebih penting daripada solusi optimum masalah tersebut. Pentingnya informasi tambahan yang dapat diperoleh dari tabel simpleks optimum telah mendorong munculnya teori dualitas dan analisis sensitivitas. Istilah dualitas menunjuk pada

kenyataan bahwa setiap program linier terdiri atas dua bentuk. Bentuk primal dan bentuk dual. Secara sistematis dualitas merupakan alat bantu masalah linier programming yang secara langsung didefinisikan dari persoalan aslinya atau dari linier programming primal.

7.1.1 Penjelasan

Istilah dualitas menunjuk pada kenyataan bahwa setiap program linier terdiri atas dua bentuk. Bentuk pertama atau bentuk asli dinamakan primal, sedangkan bentuk kedua yang berhubungan dinamakan dual sehingga suatu solusi terhadap program linier yang asli juga memberikan solusi pada bentuk dualnya. Jadi, jika suatu program linier diselesaikan dengan metode simpleks maka diperoleh penyelesaian untuk dua masalah program linier.

Perlu diingat bahwa rumus persoalan program linier terdiri dari primal dan dual. Pemecahan persoalan primal sekaligus juga bisa membantu menghitung pemecahan dual yang dikehendaki, dan juga sebaliknya. Karena dengan persoalan primal dikerjakan atau dibentuk dalam bentuk maksimalisasi, sedangkan dual dalam minimisasi dan membentuk sebaliknya.

Salah satu kunci penggunaan teori dualitas terletak pada interpretasi dan implementasi analisis sensitivitas. Karena sebagian besar nilai parameter yang digunakan dalam model asli hanyalah perkiraan kondisi masa depan dan juga akan mewakili keputusan manajerial.

Dalam kebanyakan pembahasan program linier, masalah dual didefinisikan untuk berbagai bentuk masalah primal, bergantung pada jenis batasan tanda dari variabel dan arti dari optimasi. Setiap permasalahan program linier mempunyai suatu program linier lain yang saling berkaitan disebut dual, sedemikian hingga permasalahan semula yang disebut primal solusinya dapat diperoleh dengan menyelesaikan permasalahan dualnya.

7.1.2 Asumsi-asumsi Dualitas

1. Untuk persoalan maksimasi, maka semua rumusan fungsi kendala mempunyai tanda lebih kecil dari pada atau sama dengan (\leq).
2. Untuk persoalan minimasi maka tanda fungsi syarat ikatannya harus lebih besar dari pada atau sama dengan (\geq). (Ingat bahwa tidak perlu semua konstanta atau nilai sebelah kanan (RHS) fungsi kendala yang bersangkutan harus selalu non- negatif dalam suatu rumusan yang berbentuk kanonik).

3. Jika suatu persoalan dalam rumusan Program Linier mempunyai fungsi kendala kesamaan (nilai RHS-nya bertanda sama dengan), maka fungsi kendalanya tersebut dapat ditukar atau diganti dengan dua fungsi lainnya, yang pertama bertanda “lebih kecil dari pada atau sama dengan (\leq)” dan yang kedua, bertanda “lebih besar daripada atau sama dengan (\geq)”. Salah satu diantara kedua fungsi kendala lain tersebut (dipilih salah satu), kemudian diambil, dan kalikan dengan (-1) untuk mendapatkan fungsi kendala yang sesuai dengan aturan yang diminta oleh bentuk kanonik tersebut.

Persoalan dual sebenarnya menggunakan parameter yang sama persis dengan masalah primal, namun di lokasi yang berbeda, seperti yang dirangkum di bawah ini.

1. Koefisien dalam fungsi tujuan dari persoalan primal adalah sisi kanan kendala fungsional dalam persoalan dual.
2. Sisi kanan dari kendala fungsional dalam persoalan primal adalah koefisien dalam fungsi tujuan dari persoalan dual.
3. Koefisien variabel dalam kendala fungsional dari persoalan primal adalah koefisien dalam kendala fungsional dari persoalan dual.

Bentuk Primal	Bentuk Dual
Memaksimumkan fungsi tujuan	Meminimumkan fungsi tujuan
Koefisien fungsi tujuan (C_j)	Nilai Kanan (NK) sebagai fungsi kendala
NK fungsi kendala primal (b_i)	Koefisien fungsi tujuan
Koefisien peubah ke- j	Koefisien kendala ke- j
Koefisien kendala ke- i	Koefisien peubah ke- i
Variabel ke- j yang positif (> 0)	Kendala ke- j dengan tanda ketidaksamaan lebih besar atau sama dengan (\geq)
Kendala ke- i yang bertanda ketidaksamaan (\leq)	Variabel ke- i yang positif (\geq)

Aturan umum menuliskan bentuk dual LP yang simetrik:

- a) Misalkan sebuah variable dual (*nonnegative*) untuk setiap kendala primal
- b) Vector baris koefisien fungsi tujuan primal diubah menjadi vector kolom konstan sisi kanan dual
- c) Vector kolom sisi kanan primal diubah menjadi vector baris koefisien fungsi tujuan dual
- d) Transpose koefisien matriks kendala primal menjadi koefisien matriks kendala dual

- e) Balik arah pertidaksamaan kendala
- f) Balik arah optimisasi, ubah minimum menjadi maksimum dan sebaliknya
Beberapa teori dualitas, yaitu:

1] Teori 1 (Weak Duality Theorem)

- Nilai fungsi tujuan masalah maksimasi (primal) untuk setiap solusi layak adalah batas bawah dari nilai minimum fungsi tujuan masalah dual
- Nilai fungsi tujuan masalah minimasi (dual) untuk setiap solusi layak adalah batas atas dari nilai maksimum fungsi tujuan masalah primal
- Jika masalah primal adalah layak dan nilai tujuannya tak terbatas, maka masalah dualnya tidak memiliki solusi layak
- Jika masalah primal adalah layak dan dual tak, maka primal tak terbatas
- Jika masalah dual adalah layak dan tak terbatas, maka masalah primalnya tak layak
- Jika masalah dual adalah layak dan primal tak layak, maka dualnya tak terbatas

2] Teori 2 (Optimality Criterion Theorem) Jika terdapat solusi layak dan pada bentuk primal dual simetrik demikian hingga nilai-nilai fungsi tujuan yang berhubungan adalah sama, maka solusi ini adalah solusi optimum terhadap masalah tersebut.

3] Teori 3 (Main Duality Theorem) Jika baik masalah primal maupun dual adalah layak, maka keduanya memiliki solusi demikian hingga nilai optimum fungsi tujuannya adalah sama.

4] Teori 4 (Complementary Slackness Theorem)

Kondisi Complementary Slackness dapat dinyatakan sebagai berikut:

- Jika suatu variable primal bernilai positif, maka kendala dual yang berhubungan akan dipenuhi sebagai suatu persamaan pada keadaan optimum (varabel atau pada kendala dual = 0) slack surplus
- Jika suatu kendala primal berupa pertidaksamaan murni pada keadaan optimum (varabel atau pada kendala dual > 0), maka slack surplus variable dual yang berhubungan harus sama dengan nol pada keadaan optimum

- Jika suatu variable dual bernilai positif, maka kendala primal yang berhubungan akan memenuhi sebagai suatu persamaan pada keadaan optimum (varabel atau pada kendala dual = 0) slack surplus
- Jika suatu kendala dual berupa pertidaksamaan murni (varabel slack atau pada kendala dual > 0), maka variable primal yang surplus berhubungan harus sama dengan nol pada keadaan optimum

7.2 Masalah Primal-Dual Asimetrik

Misalkan suatu masalah primal yang tak simetrik seperti berikut: Maksimumkan, Dengan kendala tak terbatas:

Ubahlah masalah di atas menjadi bentuk simetri. Ini berarti semua kendala harus menjadi pertidaksamaan jenis \leq (karena bentuk primal adalah masalah maksimasi), dan semua variable nonnegative. Ini dicapai dengan cara seperti berikut:

1. Kendala pertidaksamaan kedua dikalikan dengan -1
2. Kendala persamaan ketiga diganti dengan suatu pasangan pertidaksamaan dan
3. Variable tak terbatas diganti dengan selisih dua variable non negative

Sehingga diperoleh bentuk dualnya: Minimumkan Dengan kendala tak terbatas. Untuk setiap program linier (simetris atau tak simetris) bentuk dual selalu memenuhi ciri-ciri sebagai berikut:

1. Elemen matriks kendala bentuk dual adalah transpose elemen kendala primal
2. Koefisien fungsi tujuan dual adalah vector sisi kanan primal
3. Vector sisi kanan dual adalah koefisien fungsi tujuan primal
4. Jika primal adalah masalah maksimasi, maka dual menjadi minimasi dan sebaliknya.

Primal	Dual
Elemen matriks kendala vector sisi kanan koefisien fungsi tujuan	Transpose elemen matriks Koefisien fungsi tujuan Vektor sisi kanan

Kendala ke- berupa persamaan tak terbatas	Variabel dual tak terbatas Kendala ke- berupa persamaan
Maksimasi	Minimasi
Kendala ke- jenis \leq Kendala ke- jenis \geq ≥ 0 ≤ 0	Variabel dual ≥ 0 Variabel dual ≤ 0 Kendala ke- jenis \geq Kendala ke- jenis \leq
II. Maksimasi	Minimasi
Kendala ke- jenis \leq Kendala ke- jenis \geq ≥ 0 ≤ 0	Variabel dual ≤ 0 Variabel dual ≥ 0 Kendala ke- jenis \leq Kendala ke- jenis \geq

7.2.1 Keuntungan Perhitungan Bentuk Dual

Karena solusi suatu masalah selalu dapat diperoleh dari solusi bentuk dualnya, maka tidak perlu merumuskan kedua bentuk. Terdapat kesepakatan bahwa sejumlah besar kendala membuat masalah perhitungan lebih gawat daripada sejumlah besar variable. Ini karena jumlah kendala menentukan banyaknya vector basis dalam solusi yang pada gilirannya menentukan ukuran matriks basis invers.

Sehingga, jika suatu masalah demikian hingga bentuk primalnya memiliki sejumlah besar kendala sementara variabelnya hanya sedikit, masalah tersebut dapat diselesaikan dengan lebih efisien jika dirumuskan dalam bentuk dual.

Contoh Soal

Minimumkan

$$z = 15x_1 + 16x_2 + 30x_3$$

Dengan Kendala

$$15x_1 + 16x_2 + 30x_3 \geq 30$$

$$x_1 + 2x_2 + 5x_3 \geq 40$$

$$x_1 + x_2 + x_3 \geq 0$$

1. Susunlah dual dari masalah primal tersebut
2. Dengan menyelesaikan masalah dual, tentukan solusi masalah primal tersebut

Jawaban

1. Susunlah dual dari masalah primal tersebut

Primal

$$\text{Meminimumkan } z = 15x_1 + 16x_2 + 30x_3$$

Dengan kendala

$$2x_1 + 2x_2 + 3x_3 \geq 30$$

$$x_1 + x_2 + 5x_3 \geq 40$$

$$x_1, x_2, x_3 \geq 0$$

$$A = \begin{pmatrix} 2 & 2 & 3 \\ 1 & 2 & 5 \end{pmatrix}$$

Dual

$$\text{Max } w = 30y_1 + 40y_2$$

$$\text{s.t. } 2y_1 + y_2 \leq 15$$

$$2y_1 + 2y_2 \leq 16$$

$$3y_1 + 5y_2 \leq 30$$

$$y_1, y_2 \geq 0$$

$$A^T = \begin{pmatrix} 2 & 1 \\ 2 & 2 \\ 3 & 5 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$$

2. Dengan menyelesaikan masalah dual, tentukan solusi masalah primal tersebut

Bentuk kanonik

$$w - 30y_1 - 40y_2 - 0s_1 - 0s_2 - 0s_3 = 0$$

$$2y_1 + y_2 + S_1 = 15$$

$$2y_1 + 2y_2 \leq 16$$

$$3y_1 + 5y_2 + S_3 = 30$$

Dual

$$\text{Max } w = 30y_1 + 40y_2$$

$$\text{s.t. } 2y_1 + y_2 \leq 15$$

$$2y_1 + 2y_2 \leq 16$$

$$y_1, y_2 \geq 0$$

Tabel 1

	Basic	W	y_1	y_2	s_1	s_2	s_3	rhs	ratio
	B_0	w	1	-30	-40	0	0	0	0
	B_1	s_1	0	2	1	1	0	15	15/1 = 15
	B_2	s_2	0	2	2	0	1	16	16/2 = 8
	B_3	s_3	0	3	5	0	0	30	30/5 = 6
$b_3' = \frac{b_3}{5}$			0	$\frac{3}{5}$	1	0	0	$\frac{1}{5}$	6
$b_0' = b_0 + 40 \cdot b_3'$			$1 + 40 \cdot 0$	$-30 + 40 \cdot \frac{3}{5}$	$-40 + 40 \cdot 1$	$0 + 40 \cdot 0$	$0 + 40 \cdot 0$	$0 + 40 \cdot \frac{1}{5}$	$0 + 40 \cdot 6$
			= 1	= -6	= 0	= 0	= 0	= 8	= 240
$b_1' = b_1 - b_3'$			$0 - 0$	$2 - \frac{3}{5}$	$1 - 1$	$1 - 0$	$0 - 0$	$0 - \frac{1}{5}$	$15 - 6$
			= 0	= $\frac{3}{5}$	= 0	= 1	= 0	= $\frac{1}{5}$	= 9
$b_2' = b_2 - 2 \cdot b_3'$			$0 - 2 \cdot 0$	$2 - 2 \cdot \frac{3}{5}$	$2 - 2 \cdot 1$	$0 - 2 \cdot 0$	$1 - 2 \cdot 0$	$0 - 2 \cdot \frac{1}{5}$	$16 - 2 \cdot 6$
			= 0	= $\frac{4}{5}$	= 0	= 0	= 1	= $\frac{3}{5}$	= 4

Tabel 2

Basic	W	y_1	y_2	s_1	s_2	s_3	rhs	ratio
B_0	w	1	-6	0	0	8	240	
B_1	s_1	0	$\frac{7}{5}$	0	1	0	9	$9/7/5 = \frac{45}{7}$
B_2	s_2	0	$\frac{4}{5}$	0	0	1	4	$9/4/5 = 5$
B_3	s_3	0	$\frac{3}{5}$	1	0	0	6	$6/3/5 = 10$
$b_2' = \frac{5}{4}b_2$		0	1	0	0	$\frac{5}{4}$	$-\frac{1}{2}$	5
$b_0' = b_0 + 6.b_2'$		$1 + 6.0$	$-6 + 6.1$	$-0 + 6.0$	$0 + 6.0$	$0 + 6.\frac{5}{4}$	$8 + 6.(-\frac{1}{2})$	$240 + 6.5$
		$= 1$	$= 0$	$= 0$	$= 0$	$= \frac{12}{2}$	$= 5$	$= 270$
$b_1' = b_1 - \frac{7}{5}.b_3'$		$0 - \frac{7}{5}.0$	$\frac{7}{5} - \frac{7}{5}.1$	$0 - \frac{7}{5}.0$	$1 - \frac{7}{5}.0$	$0 - \frac{7}{5}.\frac{5}{4}$	$-\frac{1}{5} - \frac{7}{5}.(-\frac{1}{5})$	$9 - \frac{7}{5}(5)$
		$= 0$	$= 0$	$= 0$	$= 1$	$= -\frac{7}{4}$	$= \frac{1}{2}$	$= 2$
$b_2' = b_1 - \frac{3}{5}.b_3'$		$0 - \frac{3}{5}.0$	$\frac{3}{5} - \frac{3}{5}.1$	$1 - \frac{3}{5}.0$	$0 - \frac{3}{5}.0$	$0 - \frac{3}{5}.\frac{5}{4}$	$\frac{1}{5} - \frac{3}{5}.(-\frac{1}{2})$	$6 - \frac{3}{5}.5$
		$= 0$	$= 0$	$= 1$	$= 0$	$= \frac{3}{4}$	$= \frac{1}{2}$	$= 3$

Tabel 3

Basic	W	y_1	y_2	s_1	s_2	s_3	rhs
w	1	0	0	0	$\frac{12}{2}$	5	240
s_1	0	0	0	1	$-\frac{7}{4}$	$\frac{7}{5}$	2 $W^* = 270$
s_2	0	1	0	0	$\frac{5}{4}$	$-\frac{2}{5}$	5 $y_1 = 5$
s_3	0	0	1	0	$-\frac{3}{4}$	$\frac{1}{5}$	3 $y_2 = 3$

Primal

Memimumkan

$$z = 15x_1 + 16x_2 + 30x_3$$

Dengan kendala

$$z^* = 15.0 + 16.\frac{15}{2} + 30.5 = 270$$

Bab 8

Teori Analisis Sensitivitas

8.1 Pendahuluan

Dalam dunia riset operasi, analisis sensitivitas menjadi salah satu alat yang tak ternilai untuk menguji batas-batas model matematis. Melalui teknik ini, kita dapat mengeksplorasi bagaimana perubahan parameter akan mempengaruhi hasil dari model yang telah kita buat. Seolah-olah kita bisa meramal masa depan hanya dengan menganalisis angka-angka yang ada.

Misalkan kita sedang mengembangkan model perencanaan produksi untuk sebuah pabrik. Dengan menerapkan analisis sensitivitas, kita dapat mengukur seberapa responsif model kita terhadap perubahan-perubahan tertentu. Ini akan membantu kita dalam membuat keputusan yang lebih bijak sebelum diterapkan di dunia nyata.

Analisis sensitivitas adalah analisa bagaimana pengaruh perubahan data terhadap solusi optimum. Solusi optimum masalah linear program didasarkan pada nilai koefisien fungsi tujuan maupun kemampuan penyediaan sumber daya, yang dapat diketahui secara pasti. Pada kenyataannya nilai koefisien fungsi tujuan maupun kemampuan penyediaan sumber daya, sangat mungkin untuk mengalami perubahan di masa yang akan datang.

Solusi optimum akan sangat dipengaruhi oleh perubahan koefisien fungsi tujuan, koefisien kendala, kapasitas kendala, penambahan kegiatan baru, karena hal ini mengubah persoalan program linear. Maka untuk menghadapi perubahan tersebut dikembangkan suatu strategi untuk menganalisis bagaimana solusi optimum berhubungan dengan perubahan data di atas.

Analisa sensitivitas dilakukan setelah solusi optimum dari masalah program linear ditemukan, baik secara grafik maupun metode simpleks. Maka informasi tabel atau grafik optimum sangat dibutuhkan untuk melakukan analisis sensitivitas, dengan demikian analisa sensitivitas berusaha menjawab seberapa jauh perubahan data yang bagus tanpa mengubah solusi optimum.

Ada alasan mengapa analisis sensitivitas sangat penting dalam riset operasi. Yaitu keputusan yang diambil berdasarkan model matematis biasanya melibatkan banyak variabel dan parameter. Oleh karena itu, kita perlu memastikan bahwa model kita cukup tangguh untuk merespon perubahan-perubahan tersebut.

8.2 Linear Programming

Menurut Subagyo (2000) program linear adalah suatu model umum yang dapat digunakan dalam pemecahan masalah pengalokasian sumber-sumber yang terbatas secara optimal. Program linear mencakup perencanaan kegiatan untuk mencapai hasil yang optimal yaitu suatu hasil yang mencerminkan tercapainya sasaran tertentu yang paling baik di antara alternatif-alternatif yang mungkin dengan menggunakan fungsi linear.

Masalah keputusan yang biasa dihadapi analis adalah alokasi optimum sumber daya yang langka yaitu modal, tenaga kerja, bahan mentah, waktu atau teknologi. Tugas analis adalah menentukan bagaimana memaksimalkan profit, penjualan, kesejahteraan dan meminimalisasikan biaya, waktu dan jarak. Setelah masalah diidentifikasi dan tujuan diterapkan langkah selanjutnya adalah formulasi 3 tahap yang meliputi:

1. Menentukan variabel keputusan dalam bentuk formulasi matematika.
2. Membentuk fungsi tujuan yang ditujukan sebagai suatu hubungan linear dari variabel keputusan.
3. Menentukan semua kendala masalah tersebut dan mengekspresikan dalam persamaan dan pertidaksamaan yang juga merupakan hubungan linear dari variabel keputusan yang mencerminkan keterbatasan sumberdaya masalah itu.

8.2.1 Analisis Sensitivitas

Analisis sensitivitas merupakan analisa akan dampak dari perubahan data mengenai fungsi tujuan, fungsi kendala, atau kapasitas kendala terhadap solusi optimum. perubahan yang mungkin dihadapi dalam analisa sensitivitas adalah sebagai berikut:

1. Perubahan koefisien fungsi tujuan.
2. Perubahan konstanta ruas kanan.
3. Perubahan fungsi kendala.

Dari berbagai masalah yang mungkin dihadapi diatas, secara umum analisa sensitivitas berusaha untuk menjawab beberapa hal sebagai berikut:

1. Berapa besar koefisien fungsi tujuan dapat berubah tanpa mengubah solusi optimum.
2. Seberapa besar koefisien variabel non basis dapat dinaikkan sehingga cukup ekonomis untuk dibuat.
3. Sumber daya manakah yang dapat dinaikkan dan seberapa besar perubahan dibolehkan, sehingga nilai Z . dapat dinaikkan akan tetapi tanpa melakukan perhitungan dari awal.
4. Sumber daya manakah yang dapat dikurang tanpa menurunkan nilai Z maupun menghitung dari awal.
5. Sumber daya manakah yang diprioritaskan
6. Apakah penambahan kendala maupun kegiatan baru akan mempengaruhi solusi optimum.

8.2.2 Kelebihan dan Kekurangan

Kelebihan:

1. Mengidentifikasi variabel yang paling berpengaruh terhadap hasil
2. Memberikan pemahaman yang lebih mendalam tentang dampak perubahan pada variabel input
3. Memungkinkan pengambilan keputusan yang lebih baik dalam menghadapi perubahan kondisi atau situasi

Kekurangan:

1. Tidak mempertimbangkan kemungkinan interaksi antar variabel input
2. Bergantung pada asumsi yang dibuat selama simulasi
3. Tidak memberikan solusi yang pasti atau akurat, tetapi hanya memberikan pendekatan atau perkiraan

Dua hal yang ditinjau dalam analisis sensitivitas:

- Sensitivitas dari solusi optimal terhadap ketersediaan sumber daya (ruas kanan kendala). Menentukan sumber daya yang bisa ditambah dan dikurang.
- Sensitivitas dari solusi optimal terhadap perubahan profit atau biaya (Koefisien fungsi Objektif). Perbandingan antara perubahan nilai Z terhadap perubahan ketersediaan sumber (Shadow Price).

8.2.3 Bagaimana Cara Melakukan Analisa Sensitivitas dalam Riset Operasi

- Identifikasi variabel input

Langkah pertama dalam melakukan analisa sensitivitas adalah mengidentifikasi variabel input yang relevan. Variabel input ini dapat berupa harga, biaya, jumlah produksi, waktu, atau lainnya yang berkaitan dengan model riset operasi yang sedang dianalisis.

- Tentukan range perubahan

Selanjutnya, tentukan rentang perubahan untuk setiap variabel input yang akan dianalisis. Rentang ini mencakup nilai minimum dan maksimum yang mungkin terjadi dalam situasi yang berbeda. Misalnya, jika variabel input adalah harga, rentang perubahan bisa mencakup kenaikan atau penurunan harga tertentu.

- Lakukan simulasi

Setelah variabel input dan range perubahan telah ditentukan, langkah selanjutnya adalah melakukan simulasi menggunakan model riset operasi. Dalam simulasi ini, setiap variabel input akan diubah nilainya sesuai dengan range perubahan yang telah ditentukan, dan hasilnya akan dicatat.

- Analisis hasil

Setelah simulasi selesai, hasil yang diperoleh akan dianalisis. Perhatikan perbedaan hasil yang muncul akibat perubahan pada variabel input. Variabel input yang memiliki pengaruh signifikan terhadap hasil akan menjadi fokus analisis sensitivitas. Dalam analisis ini, variabel input yang memiliki pengaruh terbesar dapat diidentifikasi dan diprioritaskan dalam pengambilan keputusan.

8.2.4 Tips Analisa Sensitivitas dalam Riset Operasi

1. Identifikasi variabel input

Langkah pertama dalam melakukan analisa sensitivitas adalah mengidentifikasi variabel input yang relevan. Variabel input ini dapat berupa harga, biaya, jumlah produksi, waktu, atau lainnya yang berkaitan dengan model riset operasi yang sedang dianalisis.

2. Tentukan range perubahan

Selanjutnya, tentukan rentang perubahan untuk setiap variabel input yang akan dianalisis. Rentang ini mencakup nilai minimum dan maksimum yang mungkin terjadi dalam situasi yang berbeda. Misalnya, jika variabel input adalah harga, rentang perubahan bisa mencakup kenaikan atau penurunan harga tertentu.

3. Lakukan simulasi

Setelah variabel input dan range perubahan telah ditentukan, langkah selanjutnya adalah melakukan simulasi menggunakan model riset operasi. Dalam simulasi ini, setiap variabel input akan diubah nilainya sesuai dengan range perubahan yang telah ditentukan, dan hasilnya akan dicatat.

4. Analisis Hasil

Hasil yang muncul akibat perubahan pada variabel input. Variabel input yang memiliki pengaruh signifikan terhadap hasil akan menjadi fokus analisis sensitivitas. Dalam analisis ini, variabel input yang memiliki pengaruh terbesar dapat diidentifikasi dan diprioritaskan dalam pengambilan keputusan.

Bab 9

Penugasan dan *Software* POM

9.1 Pendahuluan

Pada era teknologi yang sudah sangat maju seperti sekarang ini, nampaknya kebutuhan penggunaan mesin tidak bisa diingkari lagi. Manusia pada umumnya memiliki pekerjaan yang harus diselesaikan untuk memenuhi kebutuhan hidupnya. Pada era teknologi seperti sekarang ini, dengan semakin beraneka ragamnya pekerjaan, tidak semua pekerjaan mampu dikerjakan secara manual atau handmade. Manusia nampaknya sangat membutuhkan alat bantu, agar pekerjaan dapat diselesaikan dengan cepat dan hasil yang optimal. Banyak alat bantu yang bisa digunakan dan salah satunya adalah mesin. Pada umumnya satu pekerjaan mampu diselesaikan dengan menggunakan beberapa mesin dan sebaliknya satu mesin dapat digunakan untuk menyelesaikan beberapa pekerjaan. Permasalahan yang kemudian muncul adalah bagaimana memilih penyelesaian pekerjaan dengan menggunakan mesin yang ada sedemikian sehingga biaya total menjadi minimum.

Metode penugasan (*assignment method*) merupakan salah satu metode yang dapat digunakan untuk mencari penugasan yang paling efisien untuk pekerjaan ke mesin atau mesin ke pekerjaan. Tujuan utama dari metode penugasan adalah meminimumkan biaya total dari pekerjaan tertentu yang dikerjakan mesin tertentu. Karakteristik utama dari model penugasan adalah satu pekerjaan ditujukan untuk satu mesin atau satu mesin ditujukan untuk satu pekerjaan. Dalam teori himpunan dikenal dengan korespondensi satu-satu atau one to one basis. Model penugasan ini merupakan suatu bentuk khusus dari linear programming. Linear programming ini sebenarnya dapat digunakan untuk mendapatkan solusi optimal dari suatu masalah penugasan, tetapi meskipun demikian, suatu algoritma yang lebih efisien telah dikembangkan untuk permasalahan penugasan ini yang dinamakan metode Hungarian.

Sedangkan tujuan yang ingin dicapai dalam menyelesaikan masalah ini adalah berusaha untuk menjadwalkan setiap assignee pada suatu assignment sedemikian rupa sehingga kerugian yang ditimbulkan minimal atau keuntungan yang didapatkan maksimal. Yang dimaksud dengan kerugian dalam masalah ini adalah biaya dan waktu, sedangkan yang termasuk dalam keuntungan di antaranya adalah pendapatan, laba dan nilai kemenangan. Dari sini terlihat bahwa secara garis besar ada dua jenis masalah pemberi tugas yaitu:

1. Maksimasi
2. Minimasi

9.2 Penjelasan

9.2.1 Metode Penugasan/Hungarian

Metode Penugasan disebut juga dengan Metode Hungarian, Metode Hungarian adalah metode yang memodifikasi baris dan kolom dalam matriks efektifitas sampai muncul sebuah komponen nol tunggal dalam setiap baris atau kolom yang dapat dipilih sebagai alokasi penugasan. Sejarah Awalnya metode Hungarian ditemukan dan dipublikasikan oleh Harold W. Kuhn pada tahun 1955 lalu diperbaiki oleh James Munkres pada tahun 1957 yang kemudian dikenal juga dengan nama algoritma Kuhn Munkres: Algoritma yang dikembangkan oleh Kuhn-Munkers tersebut didasarkan pada hasil kerja dua orang matematikawan asal Hungaria lainnya, yaitu Denes Konig dan Jenő Egervary. Secara umum langkah-langkah penyelesaian masalah penugasan yang normal adalah:

1. Identifikasi dan Penyederhanaan Masalah.
2. Dibuat dalam bentuk tabel penugasan.

- » Minimalkan

Untuk kasus minimalisasi, maka perlu mencari nilai terkecil setiap baris, kemudian menggunakan nilai terkecil tersebut untuk mengurangi semua nilai yang ada pada baris yang sama.

- » Maksimalkan

Untuk kasus maksimalisasi, maka perlu mencari nilai tertinggi setiap baris, kemudian nilai tertinggi tersebut dikurangi dengan nilai-nilai yang ada pada baris yang sama.

3. Memastikan Semua Baris dan Kolom Sudah Memiliki Nilai Nol.

Apabila masih ada baris atau kolom yang belum memiliki nilai nol, maka dicari nilai terkecil pada baris/kolom tersebut untuk kemudian digunakan untuk mengurangi semua nilai yang ada pada baris/kolom tersebut.

4. Memastikan Tidak Terjadi Bentrok Pada Nilai Nol.

Dengan kata lain, apakah nilai nol (yang mewakili penugasan) mengalami bentrok atau menjadi rebutan bagi sumber daya lain? jika iya, maka masih perlu dioptimalkan. Setelah semua baris dan kolom memiliki nilai nol, maka langkah selanjutnya adalah memastikan atau mengecek apakah dalam tabel penugasan tersebut telah berhasil ditemukan nilai nol sebanyak sumber daya (bisa karyawan, mesin, alat transportasi, dll.) yang juga tercermin dengan jumlah barisnya. Misalnya jika yang akan ditugaskan adalah 4 karyawan, maka harus ditemukan nilai nol sebanyak 4 buah yang terletak di baris dan kolom yang berbeda. Sebaiknya dimulai dari baris yang hanya memiliki satu nilai nol. Langkah ini mengandung arti bahwa setiap karyawan hanya dapat ditugaskan pada satu pekerjaan saja.

5. Menarik Garis Yang Menghubungkan Nol. Bila belum, maka langkah selanjutnya adalah menarik garis yang menghubungkan minimal dua buah nilai nol dalam tabel penugasan tersebut.

6. Mengurangi Nilai Di luar Garis dan Menambah Nilai Di dalam Garis. Selanjutnya, perhatikan nilai-nilai yang belum terkena garis. Pilih nilai yang paling kecil, kemudian pergunakan untuk mengurangi nilai-nilai yang belum terkena garis, dan gunakan untuk menambah nilai-nilai yang terkena garis ganda (dua kali)

7. Sudahkan Optimal? Dari hasil langkah ke 6 tersebut, apakah sudah didapatkan nilai nol sejumlah sumber daya?, yang juga tercermin dengan jumlah barisnya. Jika sudah, maka masalah penugasan telah optimal, tapi jika belum, maka silakan ulangi langkah penyelesaian ke 5.

Sebagai catatan, kasus penugasan dianggap normal bila jumlah sumber daya yang akan ditugaskan dan jumlah pekerjaan atau tujuan sama.

9.2.2 Analisis POM

Program POM merupakan sebuah program komputer yang digunakan untuk memecahkan masalah manajemen dalam bidang produksi dan operasi yang bersifat kuantitatif. Kemudahan pengoperasian menjadikan POM for Windows sebagai alternatif aplikasi guna membantu pengambilan keputusan seperti misalnya menentukan kombinasi produksi yang sesuai agar memperoleh keuntungan sebesar-besarnya. Menentukan order pembelian barang agar biaya perawatan menjadi seminimal mungkin, menentukan penugasan karyawan terhadap suatu pekerjaan agar dicapai hasil yang maksimal, dan lain sebagainya. Program POM dalam aplikasinya di komputer menyediakan beberapa jenis modul berbeda, di antaranya yaitu:

1. Aggregate Planning
2. Assignment (Penugasan)
3. Balancing Assembly Line
4. Break Even/Cost-Volume Analysis
5. Decision Analysis (Pengambilan Keputusan)
6. Forecasting (Peramalan)
7. Inventory (Persediaan)
8. Job Shop Sceduling
9. Learning Curve
10. Linnier Programing (Pemrograman Linier)
11. Location Pedoman Praktikum POM for Windows
12. Lot Sizing
13. Material Requirements Planning
14. Operations Layout
15. Project Management (PERT/CPM)
16. Quality Control
17. Reliability
18. Simulation
19. Transportation
20. Waiting Lines (Antrian)

Beberapa dari modul tersebut digunakan juga secara luas dalam manajemen operasional di fasilitas kesehatan untuk membantu direktur rumah sakit maupun kepala klinik atau puskesmas dalam mengambil keputusan manajerial.

1. Menjalankan POM for Windows

Untuk menjalankan program POM for Windows 3, dapat dilakukan:

- a) Melalui Shortcut: apabila ada shortcut POM for Windows maka klik 2x pada icon (Gambar) Shortcut POM for Windows.
- b) Melalui Menu Program: Klik start → Program → Pilih POM for Windows sehingga akan muncul layar berikut:



Secara garis besar layar POM for Windows terdiri atas:

1. Title Bar

Terdiri dari: The control Main Box, program name dan button untuk layar yaitu Minimize, Maximize, dan close.

2. Menu Bar

Terdiri dari: File, Edit, View, Modul, Tables, Tools, Windows, dan Help

3. Tool Bar atau Button Bar

Terdiri dari: Command Bar, contohnya print screen dan solve, Instruction Panel, Extra Data Area, Data Table, Annotation Area, Status Panel.

4. Menyimpan File Pada POM for Windows

Ketika hendak menyimpan file maka tutup semua hasil yang dibuka melalui tombol WINDOW. Kemudian kembali ke data awal dengan menekan tombol edit data sehingga muncul tabel. Apabila grafik tidak ditutup dulu maka akan muncul pertanyaan save as bmp file? Berikut ekstensi yang harus diperhatikan ketika menyimpan file pada MODULE yang akan dibahas:

- a) Linnier Programming save as file dengan ekstensi **.lin**
- b) Transportations save as file dengan ekstensi **.tra**
- c) Assignment save as file dengan ekstensi **.ass**
- d) Break-even / Cost Volume Analysis save as file dengan ekstensi **.bre**
- e) Inventory save as file dengan ekstensi **.inv**

9.2.3 Modul Penugasan (Assignment Modul)

Modul ini digunakan untuk memecahkan masalah-masalah yang berkaitan dengan assignment atau penugasan, seperti penugasan karyawan pada jenis pekerjaan tertentu atau penugasan karyawan dengan mesin tertentu dengan tujuan memaksimalkan keuntungan atau meminimalkan biaya.

9.3 Masalah Penugasan

9.3.1 Metode Penugasan/Hungarian

Fase 1:

- Cari nilai minimum untuk tiap-tiap baris. Kurangkan setiap entri dari suatu baris dengan nilai minimum masing-masing baris.
- Cari nilai minimum untuk tiap-tiap kolom. Kurangkan setiap entri dari suatu kolom dengan nilai minimum masing-masing kolom.

Contoh: Terdapat 5 pekerjaan dan 5 operator. Tabel berikut adalah waktu yang diperlukan oleh masing-masing operator untuk menyelesaikan pekerjaan. Tentukan bagaimana cara menugaskan operator ke pekerjaan yang sesuai sedemikian sehingga waktu penyelesaian pekerjaan seminimum mungkin.

		Operator					Nilai Minimum Baris
		1	2	3	4	5	
Pekerjaan	1	10	12	15	12	6	6
	2	5	14	12	12	11	5
	3	12	14	5	7	9	5
	4	12	8	11	12	10	8
	5	7	13	12	11	15	7

Setelah masing-masing baris dikurangkan dengan nilai minimum baris, hasilnya adalah sebagai berikut:

		Operator				
		1	2	3	4	5
Pekerjaan	1	4	6	9	6	0
	2	0	9	7	7	6
	3	7	9	0	5	4
	4	4	0	3	4	2
	5	0	6	5	4	8
Nilai minimum kolom		0	0	0	2	0

Setelah masing-masing kolom dikurangkan dengan nilai minimum kolom, hasilnya adalah sebagai berikut:

		Operator				
		1	2	3	4	5
Pekerjaan	1	4	6	9	4	0
	2	0	9	7	5	6
	3	7	9	0	0	4
	4	4	0	3	2	2
	5	0	6	5	2	8

Fase 2: Optimisasi dari Problem

- Langkah 1: Tarik garis dengan jumlah minimum untuk menutupi semua angka nol dari matriks (tabel). Prosedur:
 - » Scan baris
 - Mulai dari baris pertama, tanyakan pertanyaan berikut, “Apakah hanya ada satu angka nol di baris tersebut?”. Jika ya, tandai angka nol tersebut dengan kotak dan tarik garis VERTIKAL melewati angka nol tersebut. Jika tidak, lewati baris tersebut.
 - Setelah melakukan scan terhadap baris terakhir, cek apakah semua angka nol telah tertutupi oleh garis. Jika ya, lanjutkan ke Langkah 2, jika tidak, lakukan scan kolom berikut.

- » Scan kolom
 - Mulai dari kolom pertama, tanyakan pertanyaan berikut, “Apakah hanya ada satu angka nol di kolom tersebut?”. Jika ya, tandai angka nol tersebut dengan kotak dan tarik garis HORIZONTAL melewati angka nol tersebut. Jika tidak, lewati kolom tersebut.
 - Setelah melakukan scan terhadap kolom terakhir, cek apakah semua angka nol telah tertutupi oleh garis.
- Langkah 2: Cek apakah jumlah tanda kotak sama dengan jumlah baris dari matriks. Jika ya, pergi ke Langkah 5. Jika tidak, pergi ke Langkah 3.
- Langkah 3: Identifikasi nilai minimum dari nilai-nilai entri yang belum tercoret (terhapus).
 - » Tambahkan entri yang berada di perpotongan baris dengan nilai minimum entri yang belum tercoret (terhapus) di matriks (tabel) saat ini.
 - » Kurangkan entri yang belum tercoret (terhapus) dengan nilai minimum entri yang belum tercoret di matriks (tabel) saat ini.
 - » Entri yang lain tetap sama.
- Langkah 4: Pergi ke Langkah 1.
- Langkah 5: Solusi optimal adalah semua entri yang ditandai dengan kotak.

		Operator				
		1	2	3	4	5
Pekerjaan	1	4	6	9	4	0
	2	0	9	7	5	6
	3	7	9	0	0	4
	4	4	0	3	2	2
	5	0	6	5	2	8

Ternyata, jumlah kotak belum sama dengan jumlah baris.

Tentukan nilai minimum entri yang belum tercoret, dalam contoh ini adalah 2.

		Operator				
		1	2	3	4	5
Pekerjaan	1	4	6	9	4	0
	2	0	9	7	5	6
	3	7	9	0	0	4
	4	4	0	3	2	2
	5	0	6	5	2	8

Jalankan Langkah 3 (entri yang tidak terhapus dikurangkan 2, entri pada perpotongan kedua garis ditambah 2), yang lain tetap.

Diperoleh matriks baru sebagai berikut:

		Operator				
		1	2	3	4	5
Pekerjaan	1	4	6	7	2	0
	2	0	9	5	3	6
	3	9	11	0	0	6
	4	4	0	1	0	2
	5	0	6	3	0	8

Jumlah kotak sudah sama dengan jumlah baris.

- Langkah 5: Solusi optimal

Pekerjaan 1 dikerjakan oleh operator 5

Pekerjaan 2 dikerjakan oleh operator 1

Pekerjaan 3 dikerjakan oleh operator 3

Pekerjaan 4 dikerjakan oleh operator 2

Pekerjaan 5 dikerjakan oleh operator 4

Total waktu: $6 + 5 + 5 + 8 + 11 = 35$

9.3.2 Analisis POM

Modul Penugasan (Assignment Modul)

Contoh Kasus

Sebuah Rumah Sakit mempunyai 4 pekerjaan yang berbeda untuk diselesaikan oleh 4 karyawan. Tabel biayanya adalah sebagai berikut:

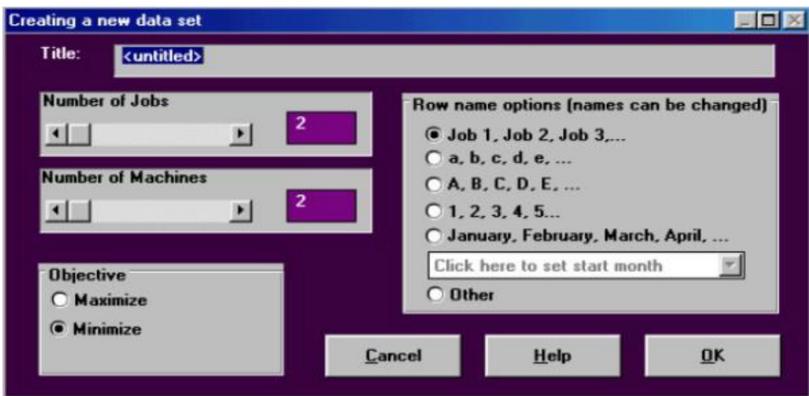
Pekerjaan Karyawan	I	II	III	IV
Raihan	150	200	180	220
Hamdan	140	160	210	170
Hasan	250	200	230	200
Dzakwan	170	180	180	160

Bagaimana menugaskan keempat karyawan untuk menyelesaikan keempat pekerjaan agar total biaya pekerjaan minimum?

Penyelesaian:

Langkah penyelesaian dengan menggunakan POM:

1. Menu utama POM, klik Modul → Assignment
2. Klik File → New untuk memasukkan data baru, akan muncul tampilan



3. Lengkapi dan isi kotak tersebut dengan data yang ada
 - » Title: Penugasan karyawan
 - » Number of jobs: jumlah pekerjaan yang ada
 - » Number of machines: jumlah karyawan yang ada

- » Objective: menggambarkan tujuan pengalokasian sumber daya
 - » Raw name options: nama batasan yang diinginkan
4. Apabila sudah terisi semua dengan benar, klik OK. Akan muncul tampilan sebagai berikut:

	I	II	III	IV
Raihan	150	200	180	220
Hamdan	140	160	210	170
Hasan	250	200	230	200
Dzakwan	170	180	180	160

5. Klik SOLVE untuk melihat hasilnya

Perhatikan ada 4 hasil proses yaitu Assignment, Marginal Cost, Original Cost, dan Assignment List. Apabila kita menginginkan keempat hasil tampil semua dapat dilakukan dengan cara klik Window → Tile

JOB	Assigned to	Cost
Raihan	III	180.
Hamdan	I	140.
Hasan	II	200.
Dzakwan	IV	160.
Total		680.

	I	II	III	IV
Raihan		40.		60.
Hamdan		10.	40.	20.
Hasan	60.		10.	
Dzakwan	20.	20.		

	I	II	III	IV
Raihan	150	200	180	220
Hamdan	140.	160.	210.	170.
Hasan	250.	200.	230.	200.
Dzakwan	170.	180.	180.	160.

	I	II	III	IV
Optimal cost = \$600				
Raihan	150	200	Assign 180	220.
Hamdan	Assign 140	160.	210.	170.
Hasan	250	Assign 200	230.	200.
Dzakwan	170	180.	180	Assign 160

Pada *assignment list* dapat kita lihat bahwa:

- » Raihan → pekerjaan III
- » Hamdan → pekerjaan I
- » Hasan → pekerjaan II
- » Dzakwan → pekerjaan IV Dengan total biayanya: Rp680

Pada Marginal Cost dapat dilihat sebagai contoh misalnya apabila Hamdan ditugaskan untuk melakukan:

- » pekerjaan II → Total cost meningkat menjadi $680 + 10 = 690$
- » pekerjaan III → Total cost meningkat menjadi $680 + 40 = 720$
- » pekerjaan IV → Total cost meningkat menjadi $680 + 20 = 700$

Daftar Pustaka

- <https://repository.uin-suska.ac.id/10403/1/Riset%20Operasi.pdf>
- https://books.google.co.id/books?hl=id&lr=&id=rwuZDwAAQBAJ&oi=fnd&pg=PR1&dq=Jurnal+tentang+perkembangan+riset+operasi&ots=9n7u3H8Tpc&sig=q9P44wuQvl3xyej1WeJ2OJMOqss&redir_esc=y#v=onepage&q&f=true
- <https://ejournal.undiksha.ac.id/index.php/JST/article>
- <https://pakdosen.co.id/riset-operasi/>
- <https://repository.ut.ac.id/3876/1/ADBI4530-M1.pdf>
- INSENTIF BAHAN AJAR RISET OPERASI-dikonversi.pdf
- Sri Mulyono. (2017). *Riset Operasi*, Mitra Wacana Media, Jakarta
- <https://www.dictio.id/t/apa-yang-dimaksud-dengan-pemrograman-linear-linear-programming-di-dalam-riset-operasi/14528/2>
- <https://id.scribd.com/document/513445303/Kelompok-1-materi-Linier-Programming-Riset-Operasi>
- <https://slideplayer.info/slide/16914839/>
- https://www.academia.edu/37872631/MAKALAH_RISET_OPERASI_Linier_Programming_Metode_Simpleks_dan_Fungsi_Tujuan_Minimum
- <https://www.dictio.id/t/apa-yang-dimaksud-dengan-pemrograman-linear-linear-programming-di-dalam-riset-operasi/14528>
- Mulyono, Sri. (2017). *Riset Operasi*. Jakarta: Mitra Wacana Media.
- Meflinda, Astuti dan Mahyarni. (2011). *Operation Research (Riset Operasi)*. Pekanbaru: UR Press.
- Hartama, Dedy dkk. (2020). *Riset Operasi: Optimalisasi Produksi Menggunakan Metode Simpleks & Metode Grafik*. Medan: Yayasan Kita Menulis.
- Setiawan Budi Tri Wahyono, Iskandar, dan R. A. (2017). Analisis Kontribusi Keuntungan Atas Produk-Produk Pada Usaha Penggilingan Daging Titipane Gusti Di Kota Bangun. *JEMI Vol 17/No 1/Januari 2017*, 17(1), 82–87.
-

- Aulia, Nila. (2015). *Makalah Kelompok 4 Metode Simpleks*. SlideShare. Diakses pada 25 September 2023, dari https://www.slideshare.net/nila_aulia/makalah-kelompok-4-metode-simpleks.
- Only, Mardillah. *Metode Simpleks Dalam Program Linier*. Academia.edu. Diakses pada 25 September 2023, dari https://www.academia.edu/11561691/METODE_SIMPLEKS_DALAM_PROGRAM_LINIER.
- Regi, Ivan Sada. *Makalah Program Linier Tentang Metode Simpleks*. Academia.edu. Diakses pada 25 September 2023, dari https://www.academia.edu/11453031/makalah_program_linier_tentang_metode_simp_leks.
- Tanenbaum, A. S., & Bos, H. (2014). *Modern operating systems*. Pearson Education.
- Smith, J. K., & Johnson, L. M. (2018). The impact of antivirus software on system performance. *Journal of Computer Security*, 25(2), 123–145.
- Microsoft Corporation. (2022). *Windows 10 Support Lifecycle*. <https://support.microsoft.com/en-us/help/13853/windows-lifecycle-fact-sheet>
- <https://ejournal.undiksha.ac.id/index.php/JPTK/article/download/24/31>
- Sri Mulyono. (2017). *Riset Operasi*, Mitra Wacana Media, Jakarta <https://www.scribd.com/doc/100066609/METODE-TRANSPORTASI> <https://prezi.com/tr4mres4acjf/metode-stepping-stone/>
- <https://www.studocu.com/id/document/universitas-negeriyogyakarta/manajemen/metode-modi- dan-vam/45143960>
- <https://text-id.123dok.com/document/eqorn8omq-jenis-jenis-model-transportasi-penyelesaian- optimasi.html>
- <https://text-id.123dok.com/document/rz3ww3dqx-asumsi-dasar-model-transportasi.html> https://spada.uns.ac.id/pluginfile.php/604968/mod_resource/content/1/14%20Masalah%20Transportasi.ppt
- Affandi, P., (2014) *Perluasan Model Kendali Optimal Sistem Pergudangan Dengan Produksi Yang Mengalami Kemosotan*. Proseding UAD, Yogyakarta.
- Affandi, P.,(2016). *Perluasan Model Kendali Optimal pada Masalah Inventori yang Mengalami Penurunan Mutu*. Proseding Seminar Nasional Matematika Udayana, ISSN 2406-9868.
- Affandi, P.,(2015). *Kendali Optimal Dari Sistem Inventori Dengan Peningkatan Dan Penurunan Barang*. *Jurnal MIPA Unnes*, Vol 38, No 1, 79-88.

- Coyle, J.J, E.J. Bardi & J.C. Langley. 2006. Management of Transportation. Thomson South-Western.
- Dantzig, G.B & M.N. Thapa. 1997. Linear Programming. SpringerVerlag. New York.
- Gunawan, Ellen, Mulia, Ardi Wirda. 1990. Pengantar Riset Operasi Edisi Kelima Jilid 1. Wahyarasmana, Dede, editor. Jakarta: Erlangga.
- Hiller S. F & J.G.Lieberman. 2010. Introduction to Operations Research. McGraw-Hill Education. New York.
- Lukmana, Tomi & Trivena, Diana. 2015. Penerapan Metode EOQ dan ROP (Studi Kasus: PD. BARU). Jurnal Teknik Informatika dan Sistem Informasi, Volume 1 Nomor 3.
- Ramakrishna, C.S. 1988. An Improvement to Goyal's Modified VAM for The Unbalanced Transportation Problem. J. Opl. Res. Society. 39: 609-610.
- Schulze, A.M. 1998. Linear Programming for Optimization: Perceptive Scientific Instruments, Inc. Pergamon Press. New York.
- Fadillah, S. (2009). KEMAMPUAN PEMECAHAN MASALAH MATEMATIS DALAM PEMBELAJARAN MATEMATIKA.
- Faigiziduhu. (2017). *OPERASI RISET PROGRAM LINIER*. Medan.
- <https://id.scribd.com/document/571066844/Bab-3-Dualitas-Dan-Analisis-Sensitivitas-Rev-1>
- https://spada.uns.ac.id/pluginfile.php/73325/mod_resource/content/4/DUALITAS..pdf
- <https://www.studocu.com/id/document/universitas-negeri-medan/operasi-riset/makalah-kelompok-6/45490234>
- Liana, Lie. (2017). Pencapaian Biaya Minimum Menggunakan Metode Hungarian Dan Daftar Kombinasi.
- https://simdos.unud.ac.id/uploads/file_pendidikan_1_dir/802963ad3ef031bd2ef76fa681672e59.pdf
- https://www.scribd.com/embeds/458892562/content?start_page=1&view_mode=scroll&access_key=key-ffexxf7r1bzEfWu3HKwf
- https://youtu.be/CNu3ikrZIA0?si=w-4BvWnq3_bzJfvU

