

BAB II

LANDASAN TEORI

2.1 Program Linier

Pemrograman matematika pada umumnya terdiri atas lebih dari satu macam hubungan variabel yang diperhatikan. Khususnya lagi dalam pemrograman matematika selalu ada bagian yang disebut fungsi tujuan (objektif) dan bagian lain yang disebut fungsi kendala (*constraints*) [4].

Program linier adalah suatu model umum yang dapat digunakan sebagai solusi masalah pengalokasian sumber-sumber yang terbatas secara optimal. Masalah tersebut timbul apabila seseorang diharuskan untuk memilih atau menentukan tingkat kegiatan yang akan dilakukannya, dimana masing-masing kegiatan membutuhkan sumber yang sama sedangkan jumlahnya terbatas [5].

Program linier merupakan kasus khusus dari pemrograman matematika. Program linier adalah metode matematika untuk menentukan cara memperoleh hasil yang optimal (keuntungan maksimum atau biaya minimum) dalam model matematika yang diberikan, yang memenuhi kondisi-kondisi tertentu yang dipresentasikan sebagai hubungan linier [6].

2.1.1 Model Program Linier

Program linier adalah suatu metode untuk membuat keputusan di antara berbagai alternatif kegiatan pada waktu kegiatan-kegiatan tersebut dibatasi oleh kendala tertentu. Keputusan yang akan diambil dinyatakan sebagai fungsi tujuan sedangkan kendala-kendala yang dihadapi dalam membuat keputusan tersebut dinyatakan dalam bentuk fungsi-fungsi kendala. Sesuai dengan nama model program linier, maka fungsi tujuan dan fungsi kendala tersebut harus berupa fungsi yang linier, baik dalam bentuk persamaan maupun pertidaksamaan pada variabel – variabel keputusannya [7].

Ada beberapa langkah-langkah memformulasikan model, yaitu:

1. Pembentukan Variabel Keputusan

Variabel keputusan adalah variabel yang akan dicari harganya dan dipilih mana yang akan memberikan hasil terbaik sesuai sasarannya yang dinyatakan dalam simbol matematik.

2. Pembentukan Fungsi Tujuan

Fungsi tujuan yang menggambarkan tujuan/sasaran di dalam permasalahan program linier yang berkaitan dengan pengaturan secara optimal sumber daya-sumber daya untuk memperoleh keuntungan maksimal atau biaya minimal yang pada umumnya nilai yang akan dioptimalkan.

3. Pembentukan Fungsi Pembatas

Meliputi penyajian matematis berupa batasan-batasan kapasitas yang tersedia dalam bentuk persamaan atau pertidaksamaan yang berupa hubungan linear dari variabel keputusan yang mencerminkan keterbatasan sumber daya yang dialokasikan secara optimal ke berbagai kegiatan.

4. Pembentukan Pembatas Tanda

Pembatas yang menjelaskan apakah variabel keputusannya diasumsikan hanya berharga non-negatif atau berharga positif boleh juga berharga negatif (tidak terbatas dalam tanda).

Secara umum model matematis permasalahan program linier adalah [8]:

Fungsi tujuan:

$$\text{Memaksimalkan } Z = c_1x_1 + c_2x_2 + c_3x_3 + \dots + c_nx_n$$

$$\text{kendala: } a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \leq b_1$$

$$a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n \leq b_2$$

.....

$$a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n \leq b_m$$

dan

$$x_1 \geq 0, x_2 \geq 0, \dots, x_n \geq 0.$$

Formulasi tersebut dinamakan sebagai bentuk standar dari persoalan program linier, dan setiap situasi yang formulasi matematisnya memenuhi model

ini adalah persoalan program linier. Selain model program linier dengan bentuk formulasi tersebut, model program linier dengan bentuk lain adalah seperti [8]:

1. Meminimumkan fungsi tujuan:

$$\text{Meminimumkan } Z = c_1x_1 + c_2x_2 + \dots + c_nx_n$$

2. Beberapa kendala fungsional dengan pertidaksamaan lebih besar atau sama dengan:

$$a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n \geq b_i, \text{ untuk beberapa nilai } i$$

3. Beberapa kendala fungsional dalam bentuk persamaan:

$$a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n = b_i, \text{ untuk beberapa nilai } i$$

4. Menghilangkan kendala-kendala non-negatif untuk beberapa variabel keputusan:

x_j tidak dibatasi dalam tanda, untuk beberapa nilai j

Misalkan:

x_j : Variabel keputusan, dengan $j = 1, 2, \dots, n$

c_i : Koefisien fungsi tujuan, dengan $i = 1, 2, \dots, n$

$a_{11}, \dots, a_{mn}$: Koefisien fungsi kendala

$b_1, b_2, \dots, b_m$: Jumlah sumber daya yang ada

2.1.2 Asumsi Program Linier

Asumsi – asumsi yang berlaku pada program linier adalah [9]:

1. *Proporsionality* (Asumsi kesebandingan)

- a. Kontribusi setiap variabel keputusan terhadap fungsi tujuan adalah sebanding dengan nilai variabel keputusan.

Pada fungsi tujuan

$$Z = c_1x_1 + c_2x_2 + c_3x_3 + \dots + c_nx_n,$$

berlaku setiap pertambahan 1 unit x_1 akan menaikkan nilai Z sebanyak c_1 , setiap pertambahan 1 unit x_2 akan menaikkan nilai Z sebanyak c_2 dan seterusnya.

- b. Kontribusi suatu variabel keputusan terhadap ruas kiri dari setiap

pembatas juga sebanding dengan nilai variabel keputusan itu.

Pada pembatas

$$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots a_nx_n \leq b_1,$$

berlaku setiap penambahan 1 unit x_1 akan menaikkan penggunaan sumber sebanyak a_{11} satuan, setiap penambahan 1 unit x_2 akan menaikkan penggunaan sumber sebanyak a_{12} satuan dan seterusnya.

2. *Additivity* (Asumsi penambahan).

Kontribusi setiap variabel keputusan terhadap fungsi tujuan bersifat tidak bergantung pada nilai dari variabel keputusan lain. Kontribusi suatu variabel keputusan terhadap ruas kiri dari setiap pembatas bersifat tidak bergantung pada nilai dari variabel keputusan yang lain.

Pada fungsi tujuan:

$$Z = 4x_1 + 8x_2$$

jika $x_1 = 12$ dan $x_2 = 7$, menghasilkan

$$Z = 48 + 56 = 104$$

Andaikan x_1 bertambah 1 unit yaitu x_1 menjadi 13, maka nilai Z menjadi $52 + 56 = 108$. Jadi nilai 4 karena kenaikan x_1 dapat langsung ditambahkan pada nilai Z mula-mula tanpa mengurangi bagian Z yang diperoleh dari x_2 . Dengan kata lain tidak ada korelasi antara x_1 dan x_2 .

3. *Divisibility* (Asumsi pembagian)

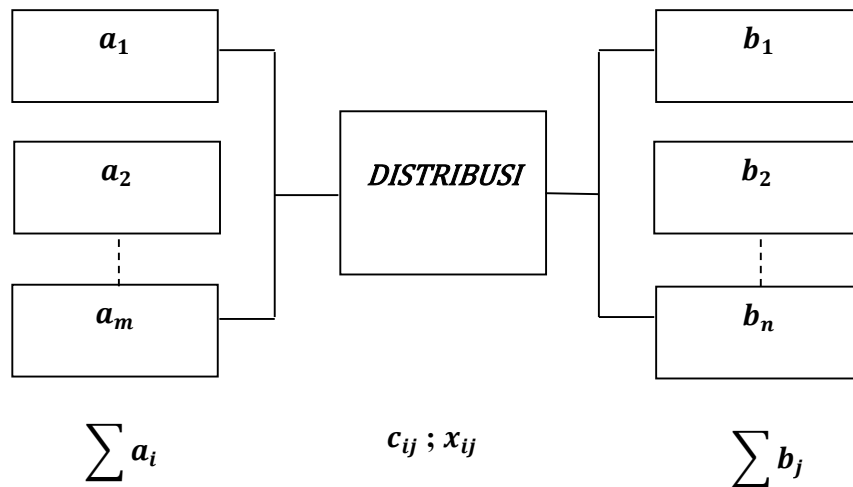
Dalam masalah program linier, variabel keputusan oleh setiap kegiatan dapat berupa bilangan pecahan. Demikian pula dengan nilai Z yang dihasilkan.

4. *Certainty* (Asumsi kepastian)

Asumsi ini menyatakan bahwa semua parameter dalam model program linier (koefisien fungsi tujuan, dan koefisien dari variabel keputusan pada pembatas) diketahui dengan jelas.

2.2 Masalah Transportasi

Kasus transportasi muncul ketika terdapat permasalahan dalam menentukan cara pengiriman (distribusi) suatu jenis barang dari beberapa sumber (lokasi penawaran) ke beberapa tujuan (lokasi permintaan) yang dapat meminimumkan biaya. Oleh karena rute pengiriman yang berbeda maka akan menghasilkan biaya pengiriman yang berbeda pula, maka tujuan dari pemecahan kasus transportasi ini adalah menentukan berapa banyak unit barang yang harus dikirim dari setiap sumber ke setiap tujuan sehingga permintaan dari setiap tujuan terpenuhi dan total biaya kirim menjadi minimum [10]. Masalah transportasi ini dapat diilustrasikan oleh Gambar 2.1 berikut [2]:



Gambar 2.1 Diagram model transportasi

Misalkan:

a_i : Volume persediaan barang sumber ke i (unit)

b_j : Volume permintaan barang dari tujuan j (unit)

c_{ij} : Biaya pengiriman per unit barang yang dikirim dari sumber i
ke tujuan j

x_{ij} : Banyaknya barang yang dikirim dari sumber i ke tujuan j (unit)

Model transportasi dapat dirumuskan sebagai berikut [2]:

Minimumkan

$$Z = \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij}$$

Dengan batasan:

$$\sum_{j=1}^n x_{ij} \leq a_i \quad ; i = 1, 2, 3, \dots, m \quad (\text{batasan persediaan})$$

$$\sum_{i=1}^m x_{ij} \leq b_j \quad ; j = 1, 2, 3, \dots, n \quad (\text{batasan permintaan})$$

$$x_{ij} \geq 0$$

Model adalah gambaran sederhana dari sebuah kasus yang dapat membantu untuk berpikir secara sistematis dan cepat untuk memahami kasus tersebut. Model transportasi menggunakan sarana sebuah tabel untuk memberikan gambaran mengenai kasus distribusi [4]. Model transportasi dapat disajikan Tabel 2.1 [2].

Tabel 2.1 Tabel transportasi

Ke Dari		Tujuan				Persediaan (supply)
		1	2	...	N	
Sumber	1	c_{11} x_{11}	c_{12} x_{12}	...	c_{1n} x_{1n}	a_1
	2	c_{21} x_{21}	c_{22} x_{22}	...	c_{2n} x_{2n}	a_2
	...	...	...	$\ddots$	...	...
	M	c_{m1} x_{m1}	c_{m2} x_{m2}	...	c_{mn} x_{mn}	a_m
Permintaan (demand)		b_1	b_2	...	b_n	

Contoh 2.1.

Studio tani adalah nama sebuah perusahaan yang menghasilkan sayuran segar khususnya hidroponik. Usaha ini berada di daerah Karanganyar, memiliki kebun

utama di desa Tagung, sedangkan kebun mitra ada di desa Jembangan dan Pabongan. Permintaan terhadap sayur hidroponik itu tidak hanya datang dari daerah sekitar, tetapi juga datang dari daerah Surakarta dan Kudus. Studio tani kemudian menunjuk tiga anggota tim untuk menjadi penanggungjawab pelayanan permintaan, dimana masing masing memegang kendali untuk daerah pemasaran/agen daerah Karanganyar, Surakarta, dan Kudus. Data rekapan permintaan ketiga daerah pemasaran tersebut untuk periode bulan yang akan datang disajikan pada Tabel 2.2:

Tabel 2.2 Tabel permintaan agen	
Daerah	Permintaan
Karanganyar	50 Kg
Surakarta	45 Kg
Kudus	55 Kg

Kemampuan berproduksi ketiga kebun itu untuk periode bulan yang akan datang disajikan pada Tabel 2.3:

Tabel 2.3 Tabel persediaan sumber	
Kebun	Kapasitas
Pabongan	40 Kg
Jembangan	50 Kg
Tagung	60 Kg

Selanjutnya, diketahui pula biaya angkut per unit dari kebun ke agen-agen, yang disajikan pada Tabel 2.4:

Tabel 2.4 Tabel biaya distribusi (rupiah/unit)

Pabrik	Agen		
	Karanganyar	Surakarta	Kudus
Pabongan	4	5	7
Jembangan	6	3	8
Tagung	5	2	3

Permasalahan Studio Tani diatas dibentuk ke dalam model transportasi:

$$\begin{aligned} \text{Minimumkan } Z &= \sum_{i=1}^3 \sum_{j=1}^3 c_{ij} x_{ij} \\ &= 4x_{11} + 5x_{12} + 7x_{13} + 6x_{21} + 3x_{22} + 8x_{23} + 5x_{31} + 2x_{32} + 3x_{33} \end{aligned}$$

dengan batasan:

$$\text{batasan persediaan: } x_{11} + x_{12} + x_{13} = 40$$

$$x_{21} + x_{22} + x_{23} = 50$$

$$x_{31} + x_{32} + x_{33} = 60$$

$$\text{batasan permintaan: } x_{11} + x_{21} + x_{31} = 50$$

$$x_{12} + x_{22} + x_{32} = 45$$

$$x_{13} + x_{23} + x_{33} = 55$$

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

Tabel Transportasinya disajikan pada Tabel 2.5.

Tabel 2.5 Tabel transportasi Studio Tani

Sumber	Tujuan			Kapasitas Sumber
	Karangayar	Surakarta	Kudus	
Pabongan	x_{11} 4	x_{12} 5	x_{13} 7	40
Jembangan	x_{21} 6	x_{22} 3	x_{23} 8	50
Tagung	x_{31} 5	x_{32} 2	x_{33} 3	60
Permintaan	50	45	55	150

Dalam perkembangannya, model transportasi telah diterapkan pada berbagai macam organisasi bisnis. Pemecahan kasus-kasus dengan model transportasi telah mengakibatkan penghematan biaya sangat efektif.

Tujuan dari model transportasi adalah merencanakan pengiriman dari sumber-sumber ke tujuan sedemikian rupa untuk meminimumkan total biaya transportasi, dengan kendala-kendala [2]:

1. Setiap permintaan tujuan terpenuhi.
2. Sumber tidak mungkin mengirim komoditas lebih besar dari kapasitasnya.

Di dalam model transportasi kemampuan sumber-sumber untuk melayani atau $\sum_{i=1}^m a_i$ belum tentu sama dengan tingkat permintaan tujuan-tujuan untuk dilayani atau $\sum_{j=1}^n b_j$. Sehingga ada tiga kemungkinan yang akan terjadi, yaitu [4]:

1. $\sum_{i=1}^m a_i = \sum_{j=1}^n b_j$.
2. $\sum_{i=1}^m a_i \leq \sum_{j=1}^n b_j$.
3. $\sum_{i=1}^m a_i \geq \sum_{j=1}^n b_j$.

Kemungkinan pertama merupakan transportasi seimbang, kemungkinan kedua dan ketiga merupakan transportasi tidak seimbang. Pada transportasi tidak seimbang harus dibuat sel *dummy* yang berisi besarnya ketidakseimbangan antara penawaran dan permintaan. Sel *dummy* dapat berupa sel baris ataupun sel kolom [11].

Metode untuk menyelesaikan masalah transportasi terdiri dari dua tahapan yaitu menentukan solusi awal atau solusi dasar transportasi yang fisibel dan menentukan solusi optimal transportasi. Penyebutan sebagai solusi awal yang fisibel dikarenakan solusi yang dihasilkan dari teknik-teknik tersebut bukanlah solusi optimal, tetapi masih layak sebagai hasil meskipun bukan hasil yang terbaik.

2.3 Masalah Penugasan

Masalah penugasan adalah kasus khusus dari masalah transportasi (*transportation problem*). Masalah umum penugasan meliputi n tugas yang harus ditetapkan kepada m pekerja dimana setiap pekerja memiliki kompetensi yang berbeda dalam menyelesaikan setiap tugasnya. Masalah penugasan (*assignment problem*) adalah suatu masalah mengenai pengaturan pada individu (objek) untuk melaksanakan tugas (kegiatan), sehingga dengan demikian biaya, waktu, jarak, ataupun keuntungan yang digunakan untuk pelaksanaan tugas tersebut dapat diminimalkan atau dimaksimalkan [5].

Dalam kasus penugasan, satu tugas atau pekerjaan dibebankan pada satu dan hanya satu objek (mesin atau orang). Ini membuat satu ciri dari kasus penugasan

dan ciri inilah yang akan membedakan kasus penugasan dari kasus transportasi. Secara umum kasus penugasan biasanya melibatkan m objek dan n tugas, x_{ij} bernilai 1 atau 0 untuk menunjukkan apakah objek i ditugaskan untuk tugas j atau tidak, c_{ij} menunjukkan biaya yang timbul dari pemberian tugas objek i pada tugas j [10].

Penggambaran umum permasalahan penugasan dapat ditulis sebagai berikut:

		Mesin				
		1	2	...	n	
Pekerjaan	1	c_{11}	c_{12}	...	c_{1n}	1
	2	c_{21}	c_{22}	...	c_{2n}	1
	.	.	.		.	.
	.	.	.		.	.
	.	.	.		.	.
	m	c_{m1}	c_{m2}	...	c_{mn}	1
		1	1	...	1	

Secara matematis, dapat dinyatakan sebagai berikut:

$$x_{ij} \begin{cases} 0, & \text{Jika pekerjaan ke-} i \text{ tidak ditugaskan pada mesin ke-} j \\ 1, & \text{Jika pekerjaan ke-} i \text{ ditugaskan pada mesin ke-} j \end{cases}$$

Dalam hal ini berlaku:

1. $x_{i1} + x_{i2} + \dots + x_{in} = 1$ untuk $i = 1, 2, \dots, m$. Ini artinya bahwa setiap objek i hanya ada satu x_{ij} yang bernilai 1 sedangkan yang lainnya bernilai 0.
2. $x_{1j} + x_{2j} + \dots + x_{mj} = 1$ untuk $j = 1, 2, \dots, n$. Ini artinya bahwa setiap tugas j hanya ada satu x_{ij} yang bernilai 1 sedangkan yang lainnya bernilai 0.

2.3.1 Persoalan Penugasan Sederhana

Persoalan penugasan sederhana adalah persoalan penugasan yang hanya memiliki satu tujuan kriteria, yaitu memaksimalkan atau meminimalkan suatu sumber daya

(biaya, waktu, kualitas atau jarak) yang digunakan untuk menyelesaikan tugas. Masalah penugasan merupakan jenis khusus pemrograman linier dimana sumber-sumber dialokasikan kepada kegiatan-kegiatan atas dasar satu - satu (*one-to-one* basis) [8].

Jadi setiap sumber atau petugas (*assignee*) seperti mesin atau karyawan ditugasi secara khusus kepada suatu kegiatan atau tugas. Ada suatu biaya c_{ij} yang berkaitan dengan petugas i ($i = 1, 2, \dots, m$) yang melakukan tugas j ($j = 1, 2, \dots, n$), sehingga tujuannya adalah untuk menentukan bagaimana semua tugas harus dilakukan untuk meminimumkan total biaya.

Jadi persoalan penugasan akan mencakup sejumlah m pekerja yang mempunyai n tugas. Dengan asumsi $n = m$, sehingga akan ada $n!$ penugasan yang mungkin dalam suatu masalah karena harus berpasangan satu-satu. Apabila pekerja i ($i = 1, 2, \dots, m$) ditugaskan kepada tugas j ($j = 1, 2, \dots, n$) maka akan muncul biaya penugasan c_{ij} , sehingga jelas bahwa tujuan dari penugasan adalah mencari penggunaan total biaya yang minimum dari semua pekerja dalam menyelesaikan semua tugas [8]. Suatu masalah umum penugasan yang hanya berkaitan dengan biaya operasi dapat direpresentasikan seperti pada Tabel 2.6.

Tabel 2.6 Matriks biaya operasi

Pekerja	Tugas						
	1	2	3	...	j	...	n
1	c_{11}	c_{12}	c_{13}	...	c_{1j}	...	c_{1n}
2	c_{21}	c_{22}	c_{23}	...	c_{2j}	...	c_{2n}
3	c_{31}	c_{32}	c_{33}	...	c_{3j}	...	c_{3n}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
I	c_{i1}	c_{i2}	c_{i3}	...	c_{ij}	...	c_{in}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
M	c_{m1}	c_{m2}	c_{m3}	...	c_{mj}	...	c_{mn}

Ada n tugas yang akan ditugaskan untuk m pekerja, c_{ij} adalah biaya operasi pekerja i untuk melaksanakan tugas j [12]. Banyak cara menyelesaikan persoalan penugasan, salah satunya adalah dengan metode *Hungarian*.

Bila pada suatu masalah ditemui adanya jumlah tugas dan pekerja berbeda (jumlah baris $\neq$ jumlah kolom), maka untuk menyamakan jumlahnya perlu ditambahkan suatu variabel semu [13], yaitu ditambahkan suatu tugas (kolom) semu jika jumlah tugas (kolom) lebih kecil daripada jumlah pekerja (baris) dan sebaliknya ditambahkan suatu pekerja (baris) semu jika jumlah pekerja (baris) lebih kecil daripada jumlah tugas (kolom). Penambahan baris ataupun kolom semu ini merupakan langkah awal dalam pembuatan tabel matriks penugasan agar dapat diselesaikan menggunakan metode *Hungarian*. Dengan demikian diasumsikan bahwa jumlah pekerja sama dengan jumlah tugas ($m = n$).

Fungsi objektif pada persoalan penugasan ini dapat ditulis sebagai berikut:

$$Z = \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \quad (2.1)$$

dengan batasan: $\sum_{j=1}^n x_{ij} = 1$, untuk $i = 1, 2, \dots, m$

$\sum_{i=1}^m x_{ij} = 1$, untuk $j = 1, 2, \dots, n$

$x_{ij} = 0$ atau $x_{ij} = 1$

$x_{ij} \begin{cases} 1, & \text{jika pekerja } i \text{ melakukan tugas } j \\ 0, & \text{jika pekerja } i \text{ tidak melakukan tugas } j \end{cases}$

dimana:

Z = Fungsi tujuan permasalahan

x_{ij} = Variabel keputusan yang menyatakan unit alokasi dari sumber i ke tujuan j

c_{ij} = Nilai kontribusi (biaya/waktu/laba) objek i terhadap tugas j

m = Jumlah objek (individu atau sumber daya)

n = Jumlah tugas yang akan diselesaikan

$x_{ij} = 1$, apabila objek i ditugaskan ke tugas j

$x_{ij} = 0$, apabila objek i tidak ditugaskan ke tugas j .

2.4 Metode *Hungarian*

Metode *Hungarian* pertama kali diperkenalkan oleh seorang ahli matematika dari *Hungarian* yang bernama D. Konig pada tahun 1916. Metode ini berhubungan dengan pemberian tugas pekerjaan terhadap karyawan untuk mengerjakan satu pekerjaan [2]. Sehingga dari pemberian tugas kepada karyawan akan diperoleh biaya optimal. Metode *Hungarian* ini mempunyai kelebihan dalam segi kesederhaan algoritma dan dari segi kemudahan untuk dipahami. Oleh karena itu metode ini merupakan pilihan para peneliti untuk menyelesaikan masalah penugasan. Berikut ini adalah persyaratan metode *Hungarian* yang harus dipenuhi yaitu:

1. Terdapat suatu permasalahan yang akan diselesaikan, bisa memaksimumkan ataupun meminimumkan.
2. Jumlah objek (m) yang ditugaskan harus sama dengan jumlah tugas (n) yang akan diselesaikan. Apabila jumlah objek tidak sama dengan jumlah tugas, atau sebaliknya, maka ditambahkan *variable dummy* terlebih dahulu.
3. Setiap objek hanya mengerjakan satu tugas.
4. Untuk dapat diselesaikan dengan menggunakan metode *Hungarian*, maka data dari masalah tersebut harus dipresentasikan dalam bentuk tabel penugasan seperti yang tersaji pada Tabel 2.7.

Tabel 2.7 Tabel penugasan

Tugas Objek	1	2	...	n
1	c_{11}	c_{12}	...	c_{1n}
2	c_{21}	c_{22}	...	c_{2n}
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
m	c_{m1}	c_{m2}	...	c_{mn}

Berikut teorema yang berkaitan dengan metode *Hungarian*, diberikan sebagai berikut:

Teorema 2.1 [14] *Jika suatu nilai konstanta ditambahkan atau dikurangkan pada semua elemen di suatu baris atau kolom matriks biaya, maka solusi optimum akan tetap sama.*

Bukti.

Diberikan a_i dan b_j , masing-masing adalah nilai konstanta yang mengurangi elemen baris ke- i dan kolom ke- j , sehingga elemen biaya c_{ij} akan berubah menjadi:

$$c'_{ij} = c_{ij} - a_i - b_j,$$

sehingga fungsi tujuan yang baru Z' adalah

$$\begin{aligned} Z' &= \sum_i \sum_j c'_{ij} x_{ij} = \sum_i \sum_j (c_{ij} - a_i - b_j) x_{ij} \\ &= \sum_i \sum_j c_{ij} x_{ij} - \sum_i a_i \left(\sum_j x_{ij} \right) - \sum_j b_j \left(\sum_i x_{ij} \right) \\ &= \sum_i \sum_j c_{ij} x_{ij} - \sum_i a_i (1) - \sum_j b_j (1) \\ &= \sum_i \sum_j c_{ij} x_{ij} - \text{Konstanta} \end{aligned}$$

Hal ini menunjukkan bahwa meminimalkan fungsi tujuan yang baru Z' menghasilkan solusi yang sama seperti meminimalkan fungsi tujuan awal Z . Walaupun nilai fungsi tujuan yang baru berbeda dengan nilai fungsi tujuan awal karena adanya pengurangan oleh suatu nilai konstanta, tetapi nilai dan posisi (letak) dari variabel x_{ij} pasti selalu sama di dalam solusi optimal kedua fungsi tujuan.

Jika dilakukan operasi pengurangan a_i dan b_j terhadap matriks biaya akan diperoleh *zero entry* atau elemen dalam matriks biaya bernilai nol yang adalah variabel yang menghasilkan solusi optimum bagi Z' . Jadi berdasarkan pembuktian, solusi optimum bagi Z' adalah solusi optimum bagi Z juga. ■

Langkah – langkah penyelesaian dengan metode *Hungarian* adalah sebagai

berikut [15]:

1.
 - a. Pada kasus minimalisasi, memilih elemen/nilai terkecil untuk setiap baris, kemudian elemen/nilai terkecil tersebut untuk mengurangi semua elemen/nilai yang ada pada baris yang sama.
 - b. Pada kasus maksimalisasi, memilih elemen/nilai terbesar untuk setiap baris dan mengurangkan elemen/nilai terbesar tersebut dengan semua elemen/nilai di baris tersebut.
2. Memastikan semua baris dan kolom sudah memiliki nilai nol. Apabila masih ada baris/kolom yang belum memiliki nilai nol, maka memilih elemen/nilai terkecil pada baris/kolom tersebut, kemudian memilih elemen/nilai terkecil tersebut untuk mengurangi semua elemen/nilai yang ada pada baris/kolom tersebut.
3. Melakukan uji optimalitas, dengan menutup elemen/nilai bernilai nol dengan garis-garis vertikal maupun horizontal pada setiap baris/kolom dengan jumlah garis sesedikit mungkin. Dalam membuat garis ini dimulai dari nolnya terbanyak. Misalkan n adalah banyaknya baris atau kolom dan l adalah banyaknya garis elemen nol, maka:
 - a. Jika $l = n$, maka penugasan sudah optimal. Proses dihentikan kemudian menentukan pasangan penugasan dari elemen nol sebagai calon solusi.
 - b. Jika $l \neq n$, maka proses dilanjutkan dengan mengikuti langkah 4.
4. Merevisi tabel penugasan, dengan memilih nilai terkecil dari elemen/nilai yang tidak tertutup garis, misalkan t , selanjutnya:
 - a. Semua elemen yang tidak tertutup garis dikurangi t .
 - b. Semua elemen yang tertutup oleh satu garis tidak diubah.
 - c. Semua elemen yang tertutup oleh dua garis ditambah dengan t .
 - d. Setelah diperoleh tabel baru kembali ke langkah 3.

Contoh 2.2.

Sebuah perusahaan kecil di bidang agrikultur memiliki enam orang karyawan

yang harus menyelesaikan enam pekerjaan yang berbeda, karena sifat pekerjaan dan juga ketrampilan, karakteristik dari masing- masing karyawan, ongkos yang timbul dari berbagai alternatif penugasan dari ke enam karyawan tersebut juga berbeda, seperti terlihat dari Tabel 2.8. Sehingga, bagaimana penugasan ini harus dibuat agar ongkos yang dikeluarkan minimal.

Tabel 2.8 Ongkos karyawan (ribu rupiah)

Karyawan	Pekerjaan					
	P_1	P_2	P_3	P_4	P_5	P_6
W_1	20	23	18	10	16	20
W_2	50	20	17	16	15	11
W_3	60	30	40	55	8	7
W_4	6	7	10	20	25	9
W_5	18	19	28	17	60	70
W_6	9	10	20	30	40	55

Berdasarkan Tabel 2.8, diperoleh data c_{ij} sebagai berikut:

$$\begin{array}{llllll}
 c_{11} = 20 & c_{12} = 23 & c_{13} = 18 & c_{14} = 10 & c_{15} = 16 & c_{16} = 20 \\
 c_{21} = 50 & c_{22} = 20 & c_{23} = 17 & c_{24} = 16 & c_{25} = 15 & c_{26} = 11 \\
 c_{31} = 60 & c_{32} = 30 & c_{33} = 40 & c_{34} = 55 & c_{35} = 8 & c_{36} = 7 \\
 c_{41} = 6 & c_{42} = 7 & c_{43} = 10 & c_{44} = 20 & c_{45} = 25 & c_{46} = 9 \\
 c_{51} = 18 & c_{52} = 19 & c_{53} = 28 & c_{54} = 17 & c_{55} = 60 & c_{56} = 70 \\
 c_{61} = 9 & c_{62} = 10 & c_{63} = 20 & c_{64} = 30 & c_{65} = 40 & c_{66} = 55
 \end{array}$$

Langkah 1

Karena kasus minimalisasi, maka memilih elemen/nilai terkecil untuk setiap baris, kemudian elemen/nilai terkecil tersebut untuk mengurangi semua elemen/nilai yang ada pada baris yang sama, hasilnya disajikan pada Tabel 2.9.

Tabel 2.9 Langkah 1 dari metode *Hungarian*

	P_1	P_2	P_3	P_4	P_5	P_6
W_1	10	13	8	0	6	10
W_2	39	9	6	5	4	0
W_3	53	23	33	48	1	0
W_4	0	1	4	14	19	3
W_5	1	2	12	0	43	53
W_6	0	1	11	21	31	46

Sebagai contoh untuk baris pertama, elemen/nilai terkecil adalah sel $c_{14} = 10$, kemudian mengurangkan elemen/nilai di baris pertama dengan $c_{14} = 10$. Diperoleh sel c_{11} menjadi $20 - 10 = 10$, c_{12} menjadi $23 - 10 = 13$, c_{13} menjadi $18 - 10 = 8$, c_{14} menjadi $10 - 10 = 0$, c_{15} menjadi $16 - 10 = 6$ dan c_{16} menjadi $20 - 10 = 10$ untuk sel pada baris yang lain sama cara pengerjaannya.

Langkah 2

Memastikan semua baris dan kolom sudah memiliki nilai nol. Pada tabel penugasan setiap baris sudah terdapat elemen/nilai nol, sedangkan pada kolom belum terdapat elemen/nilai nol pada kolom ke dua, ke tiga dan ke lima sehingga dilakukan reduksi kolom. Sebagai contoh pada kolom ke dua dilakukan reduksi kolom, elemen/nilai terkecil pada kolom ke dua adalah sel $c_{42} = 1$, kemudian mengurangkan setiap elemen/nilai pada kolom ke dua dengan elemen/nilai terkecil $c_{42} = 1$.

Diperoleh sel c_{12} menjadi $13 - 1 = 12$, c_{22} menjadi $9 - 1 = 8$, c_{32} menjadi $23 - 1 = 22$, c_{42} menjadi $1 - 1 = 0$, c_{52} menjadi $2 - 1 = 1$ dan c_{62} menjadi $1 - 1 = 0$, yang disajikan pada Tabel 2.10 berikut.

Tabel 2.10 Langkah 2 dari metode *Hungarian*

	P_1	P_2	P_3	P_4	P_5	P_6
W_1	10	12	4	0	5	10
W_2	39	8	2	5	3	0
W_3	53	22	29	48	0	0
W_4	0	0	0	14	18	3
W_5	1	1	8	0	42	53
W_6	0	0	7	21	30	46

Langkah 3

Melakukan uji optimalitas, dengan menutup elemen/nilai bernilai nol dengan garis-garis vertikal maupun horizontal pada setiap baris/kolom dengan jumlah garis sesedikit mungkin, yang disajikan pada Tabel 2.11.

Tabel 2.11 Langkah 3 dari metode *Hungarian*

	P_1	P_2	P_3	P_4	P_5	P_6
W_1	10	12	4	0	5	10
W_2	39	8	2	5	3	0
W_3	53	22	29	48	0	0
W_4	0	0	0	14	18	3
W_5	1	1	8	0	42	53
W_6	0	0	7	21	30	46

Terdapat lima garis ($l = 5$) yang dapat dibuat, dengan banyaknya baris/kolom adalah enam ($n = 6$) maka tabel penugasan belum optimal ($l \neq n$). Jadi, perlu melakukan langkah berikut:

Memilih elemen/nilai terkecil yang tidak terkena garis yaitu $t_{51} = 1$ dan menambahkan sel yang tertutup dua garis dengan t_{51} . Semua elemen yang tidak tertutup garis dikurangi 1, elemen yang tertutup garis tidak berubah dan elemen yang tertutup dua garis ditambah 1. Kemudian kembali ke langkah 3 dan

diperoleh Tabel 2.12.

Tabel 2.12 Revisi tabel penugasan

	P_1	P_2	P_3	P_4	P_5	P_6
W_1	9	11	3	0	4	10
W_2	38	7	1	5	2	0
W_3	53	22	29	49	0	1
W_4	0	0	0	15	18	4
W_5	0	0	7	0	41	53
W_6	0	0	7	22	30	47

Terdapat enam garis ($l = 6$) yang dapat dibuat, dengan banyaknya baris/kolom adalah empat ($n = 6$) maka tabel penugasan sudah optimal ($l = n$). Proses dihentikan kemudian menentukan pasangan penugasan dari elemen bernilai nol sebagai calon solusi (diberi arsiran), hasilnya disajikan pada Tabel 2.13 sebagai berikut:

Tabel 2.13 Penugasan akhir

	P_1	P_2	P_3	P_4	P_5	P_6
W_1	9	11	3	0	1	6
W_2	42	11	5	9	3	0
W_3	56	25	32	52	0	0
W_4	0	0	0	15	15	0
W_5	0	0	6	0	38	49
W_6	0	0	7	22	27	43

Selanjutnya melakukan penghitungan ongkos minimum dari penugasan yang telah dibuat, diperoleh Tabel 2.14 sebagai berikut:

Tabel 2.14 Distribusi penugasan

Karyawan	Pekerjaan					
	P_1	P_2	P_3	P_4	P_5	P_6
W_1	20	23	18	10	16	20
W_2	50	20	17	16	15	11
W_3	60	30	40	55	8	7
W_4	6	7	10	20	25	9
W_5	18	19	28	17	60	70
W_6	9	10	20	30	40	55

Sehingga ongkos penugasan yang optimal adalah (ribuan rupiah):

$$\begin{aligned}
 Z &= \sum_{i=1}^6 \sum_{j=1}^6 c_{ij}x_{ij} \\
 &= 20x_{11} + 23x_{12} + 18x_{13} + 10x_{14} + 16x_{15} + 20x_{16} + 50x_{21} + \\
 &\quad 20x_{22} + 17x_{23} + 16x_{24} + 15x_{25} + 11x_{26} + 60x_{31} + 30x_{32} + \\
 &\quad 40x_{33} + 55x_{34} + 8x_{35} + 7x_{36} + 6x_{41} + 7x_{42} + 10x_{43} + 20x_{44} + \\
 &\quad 25x_{45} + 9x_{46} + 18x_{51} + 19x_{52} + 28x_{53} + 17x_{54} + 60x_{55} + \\
 &\quad 70x_{56} + 9x_{61} + 10x_{62} + 20x_{63} + 30x_{64} + 40x_{65} + 55x_{66} \\
 &= 20(0) + 23(0) + 18(0) + 10(1) + 16(0) + 20(0) + 50(0) + \\
 &\quad 20(0) + 17(0) + 16(0) + 15(0) + 11(1) + 60(0) + 30(0) + \\
 &\quad 40(0) + 55(0) + 8(1) + 7(0) + 6(0) + 7(0) + 10(1) + 20(0) + \\
 &\quad 25(0) + 9(0) + 18(0) + 19(1) + 28(0) + 17(0) + 60(0) + \\
 &\quad 70(0) + 9(1) + 10(0) + 20(0) + 30(0) + 40(0) + 55(0) \\
 &= 10 + 11 + 8 + 10 + 19 + 9 \\
 &= 67
 \end{aligned}$$

Dengan distribusi penugasan yang diberikan adalah sebagai berikut:

Karyawan 1 mengerjakan pekerjaan 4 dengan nilai ongkos Rp. 10.000.

Karyawan 2 mengerjakan pekerjaan 6 dengan nilai ongkos Rp. 11.000.

Karyawan 3 mengerjakan pekerjaan 5 dengan nilai ongkos Rp. 8.000.
 Karyawan 4 mengerjakan pekerjaan 3 dengan nilai ongkos Rp. 10.000.
 Karyawan 5 mengerjakan pekerjaan 2 dengan nilai ongkos Rp. 19.000.
 Karyawan 6 mengerjakan pekerjaan 1 dengan nilai ongkos Rp. 9.000.
 Sehingga ongkos minimumnya adalah Rp. 67.000.

Penjelasan mengenai kasus minimalisasi dan maksimalisasi metode *Hungarian* telah menjelaskan bahwa kasus optimasi pada permasalahan penugasan tidak hanya tentang optimasi maksimasi yang biasa digambarkan dengan tujuan pencapaian laba atau keuntungan sebanyak-banyaknya, namun juga bisa berkaitan tentang penghematan biaya atau dalam kasus ini termasuk dalam kasus pengoptimalan minimalisasi biaya.

2.5 Metode *Weighted-sum*

Metode *Weighted-sum* adalah mengubah fungsi multiobjektif/multitujuan menjadi satu fungsi tujuan dengan memberikan bobot pada masing-masing fungsi objektif secara skalar.

Misal α_k menyatakan bobot fungsi tujuan ke- k . Oleh karena terdapat n fungsi tujuan, maka

$$\alpha_1 + \alpha_2 + \alpha_3 + \dots + \alpha_n = 1$$

$$\text{atau } \sum_{k=1}^n \alpha_k = 1, \quad \forall k = 1, 2, 3, \dots, n.$$

$$\alpha_i > 0, \quad i = 1, 2, \dots, k$$

dimana :

α_k = bobot fungsi tujuan ke- k

n = jumlah fungsi tujuan

Sebagai ilustrasi, model optimasi masalah penugasan multiobjektif dengan metode tersebut adalah sebagai berikut.

Meninimumkan :

$$Z = \alpha_1 \sum_{i=1}^m \sum_{j=1}^n c_{1ij} x_{ij} + \alpha_2 \sum_{i=1}^m \sum_{j=1}^n c_{2ij} x_{ij} + \cdots + \alpha_k \sum_{i=1}^m \sum_{j=1}^n c_{kij} x_{ij}$$

terhadap:

$$\sum_{i=1}^m x_{ij} = 1, \quad i = 1, 2, \dots, m$$

$$\sum_{j=1}^n x_{ij} = 1, \quad j = 1, 2, \dots, n$$

$$\alpha_1 + \alpha_2 + \alpha_3 + \cdots + \alpha_n = 1,$$

$$\alpha_i > 0, \quad i = 1, 2, \dots, k,$$

$$x_{ij} \in \{0,1\}. [3]$$