

MANAJEMEN OPERASIONAL II

Sesi Perkuliahan 2 (Sabtu, 9 Mei 2026)

Dr. Mustangin Amin, S.E., M.M.

ASSIGNMENT PROBLEM

(Mengulang dan Meneruskan Materi Sesi Perkuliahan I)

1.1. Pengantar

Masalah penugasan (*assignment problem*) merupakan suatu kasus khusus dari masalah *linear programming*. Dalam dunia bisnis dan industri manajemen sering menghadapi masalah-masalah yang berhubungan dengan penugasan optimal dan bermacam-macam sumber yang produktif atau personalia yang mempunyai tingkat efisiensi yang berbeda-beda untuk tugas-tugas yang berbeda-beda pula.

Dalam melakukan alokasi karyawan pada tugas yang ada, kadang-kadang memerlukan pemikiran yang cukup sulit. Hal ini disebabkan karena kita memiliki beberapa macam pekerjaan yang berbeda-beda cara menyelesaikannya, di samping itu karyawan yang ada memiliki keahlian dan sifat yang berbeda-beda. Alokasi karyawan ini tidak boleh asal dilakukan, sebab kalau cara alokasinya berbeda akan membawa konsekuensi hasil atau pengurbanan yang berbeda pula. Karyawan harus dialokasikan secara optimal, artinya kalau memakan biaya/pengurbanan kita usahakan sekecil-kecilnya, dan kalau menghasilkan manfaat kita usahakan sebesar-besarnya.

Cara alokasi karyawan ini dilakukan dengan menggunakan algoritma. Metode Hungarian (*Hungarian method*) adalah salah satu dari beberapa teknik-teknik pemecahan yang tersedia untuk masalah-masalah penugasan. Untuk dapat menerapkan metode Hungarian jumlah sumber-sumber yang ditugaskan harus sama dengan jumlah tugas yang akan diselesaikan. Selain itu setiap sumber yang ditugaskan hanya untuk satu tugas. Jadi masalah penugasan akan mencakup sejumlah sumber yang mempunyai n tugas. Ada $n!$ (n factorial) penugasan yang mungkin dalam suatu masalah karena perpasangan satu-satu. Masalah ini dapat dijelaskan dengan mudah oleh bentuk matriks segi empat, di mana baris-barisnya menunjukkan sumber-sumber dan kolom-kolomnya menunjukkan tugas-tugas.

Algoritma yang digunakan dalam memecahkan masalah penugasan ada dua macam, yaitu algoritma dengan meminimumkan pengurbanan (masalah minimisasi), dan algoritma dengan tujuan memaksimumkan manfaat (masalah maksimisasi).

Secara matematik masalah penugasan (*assignment problem*) dapat dinyatakan dalam suatu bentuk *linear programming* sebagai berikut:

Minimumkan (maksimumkan):

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

dengan batasan-batasan:

$$\sum_{i=1}^n X_{ij} = 1_j$$

dan

$$X_{ij} \geq 0 \quad (X_{ij} = X_{ij}^2)$$

di mana C_{ij} adalah tetapan yang sudah diketahui.

1.2. Masalah Minimisasi

Contoh: Suatu perusahaan mempunyai tiga jenis pekerjaan yang berbeda untuk diselesaikan oleh tiga karyawan, di mana karyawan tersebut mempunyai tingkat keterampilan, pengalaman kerja, latar belakang pendidikan dan latihan yang berbeda pula. Karena sifat pekerjaan dan kemampuan karyawan berbeda, maka biaya penyelesaian pekerjaan berbeda-beda. Dalam hal ini berarti ada 3! (3x2x1=6) kemungkinan penugasan.

Langkah-langkah pemecahannya adalah sebagai berikut:

1. Langkah pertama ini dimulai dengan merubah matriks biaya menjadi matriks *opportunity cost*, yaitu dengan memilih elemen terkecil pada setiap baris dari matriks biaya untuk mengurangi seluruh elemen (bilangan) pada setiap baris. Sehingga paling sedikit akan diperoleh satu bilangan yang bernilai nol pada setiap baris yang telah dikurangi bilangan terkecil pada setiap baris, di mana hal ini disebut *reduced cost matrix* seperti dalam tabel berikut:

Tabel
Cost matrix (ribuan rupiah)

Karyawan	P e k e r j a a n		
	I	II	III
A	20	27	30
B	10	18	16
C	14	16	12

Langkah selanjutnya adalah memilih bilangan terkecil pada setiap kolom dalam *reduced cost matrix* untuk mengurangi seluruh bilangan dalam kolom tersebut, sehingga menunjukkan bahwa pada setiap kolom terdapat paling sedikit satu bilangan nol.

2. *Test for formality* (menentukan apakah penugasan sudah optimal). Setiap karyawan harus ditugaskan hanya oleh satu pekerjaan dengan *oppurtunity cost* nol. Pedoman praktis untuk melakukan tes optimalisasi adalah dengan menarik garis horizontal dan/atau vertikal, dan tidak diagonal untuk meliputi seluruh bilangan bernilai nol dalam *total oppurtunity cost matrix*. Bila jumlah garis sama dengan jumlah baris atau kolom, penugasan optimal sudah tercapai. Bila tidak sama matriks harus direvisi.

a. *Reduced cost matrix* (ribuan rupiah)

Karyawan	P e k e r j a a n		
	I	II	III
A	0	7	10
B	0	8	6
C	2	4	0

b. *Total opportunity cost matrix* (ribuan rupiah)

Karyawan	P e k e r j a a n		
	I	II	III
A	0	3	10
B	0	4	6
C	2	0	0

Aplikasi tes ini pada *total opportunity cost matrix* menunjukkan bahwa penugasan optimal belum tercapai pada tahap ini. Untuk meliput seluruh bilangan nol dalam *total opportunity cost matrix* hanya memerlukan dua garis (baris C dan kolom I), sedangkan jumlah baris dan kolom adalah 3.

c. *Test for optimality* (ribuan rupiah)

Karyawan	P e k e r j a a n		
	I	II	III
A	0	3	10
B	0	4	6
C	2	0	0

3. Merevisi *total opportunity cost matrix*. Langkah ini dapat dilakukan dengan prosedur yang terdiri dari:

- a. Memilih bilangan terkecil yang tidak terliput garis-garis (yaitu *opportunity cost* terendah, atau dalam contoh kita = 3) untuk mengurangi seluruh bilangan yang tidak terliput.
- b. Menambahkan dengan jumlah yang sama (nilai bilangan terkecil) hanya pada bilangan dalam dua garis peliput yang saling bersilangan (dalam contoh ini bilangan 2 ditambah 3, atau sama dengan 5). Masukkan nilai revisi ini ke dalam matriks, sehingga didapat *total opportunity cost matrix* yang telah direvisi. Kemudian diulangi lagi langkah kedua untuk melakukan test optimalisasi. Aplikasi test langkah kedua pada revisi *total opportunity cost matrix* menunjukkan bahwa jumlah garis minimum yang diperlukan untuk meliput seluruh

bilangan nol adalah 3. Karena jumlah baris atau kolom matriks ini juga 3, maka penugasan bisa dibuat.

(1). *Revised total opportunity cost matrix* (ribuan rupiah)

Karyawan	P e k e r j a a n		
	I	II	III
A	0	0	7
B	0	1	3
C	5	0	0

(2). *Test for optimality* (ribuan rupiah)

Karyawan	P e k e r j a a n		
	I	II	III
A	0	0	7
B	0	1	3
C	5	0	0

↓

Terlihat \sum garis peliput = \sum baris/kolom, jadi matriks penugasan telah optimal, maka dapat dibuat penugasan optimal kepada masing-masing karyawan. Jadi dapat dimiliki skedul penugasan optimal dan biaya minimum sebagai berikut:

Skedul penugasan	Biaya
A	Rp 27.000,-
B	Rp 10.000,-
C	Rp 12.000,-
	Rp 49.000,-

1.3. Masalah Maksimisasi

Dalam masalah maksimisasi, matriks elemen-elemen menunjukkan tingkat keuntungan (indeks produktivitas). Efektifitas pelaksanaan tugas oleh karyawan secara individual diukur dengan jumlah kontribusi keuntungan.

Matriks kontribusi laba (ribuan rupiah)

Karyawan	P e k e r j a a n			
	I	II	III	IV
A	12	14	12	10
B	16	12	11	17
C	11	10	9	10
D	15	17	10	18

Prosedur pemecahan masalah maksimisasi dimulai dengan merubah matriks kontribusi laba menjadi matriks *opportunity loss*. Seperti sebelumnya setiap baris akan berisi paling sedikit satu bilangan nol.

a. *Opportunity Loss Matriks*

Karyawan	P e k e r j a a n			
	I	II	III	IV
A	2	0	2	4
B	1	5	6	0
C	0	1	2	1
D	3	1	8	0

b. Total Opportunity Loss Matriks

Karyawan	P e k e r j a a n			
	I	II	III	IV
A	2	0	0	4
B	1	5	4	0
C	0	1	0	1
D	3	1	6	0

Kemudian dilakukan test optimasi untuk matriks *total opportunity loss* dengan cara yang sama seperti pada masalah minimisasi. Tes ini menunjukkan bahwa seluruh bilangan nol dapat diliput hanya dengan tiga garis, sedangkan jumlah baris atau kolom adalah empat. Ini berarti matriks harus direvisi dengan cara seperti yang telah dibahas.

c. Revised Total Opportunity Loss Matrix dan Test for Optimality

Karyawan	P e k e r j a a n			
	I	II	III	IV
A	2	0	0	5
B	0	4	3	0
C	0	1	0	2
D	2	0	5	0

Skedul penugasan optimal dan kontribusi laba total untuk dua alternatif penyelesaian adalah:

Skedul Penugasan I	Kontribusi Laba	Skedul Penugasan II	Kontribusi laba
A - II	Rp 15.000,-	A - III	Rp 12.000,-
C - III	Rp 9.000,-	B - IV	Rp 17.000,-
B - I	Rp 16.000,-	C - I	Rp 11.000,-
D - IV	Rp 18.000,-	D - III	Rp 17.000,-
	Rp 57.000,-		Rp 57.000,-

1.4. Jumlah Karyawan Tidak Sama Dengan Jumlah Pekerjaan

Dalam praktek sering dijumpai kasus di mana jumlah karyawan tidak sama dengan jumlah pekerjaan. Untuk memenuhi persyaratan suatu matriks yang empat bujur sangkar, dalam hal jumlah pekerjaan lebih besar daripada jumlah karyawan, harus ditambahkan karyawan semu/boneka (*dummy worker*). Sebaliknya bila jumlah pekerjaan lebih besar daripada jumlah karyawan, harus ditambahkan pekerjaan semu (*dummy job*). Biaya penugasan untuk karyawan atau pekerjaan semu ini sama dengan nol, karena tidak akan terjadi biaya bila suatu pekerjaan ditugaskan pekerjaan semu, atau karyawan diberi tugas mengerjakan pekerjaan semu.

Contoh: Jumlah pekerjaan lebih besar daripada jumlah karyawan

Karyawan	P e k e r j a a n			
	I	II	III	IV
A	20	27	30	32
B	10	18	16	20
C	14	16	12	15
<i>Dummy</i>	0	0	0	0

Prosedur pemecahan masalah selanjutnya sama dengan langkah-langkah seperti sebelumnya.