BAB II

KAJIAN PUSTAKA

A. Efektivitas

Efektivitas berasal dari kata efektif, yang merupakan kata serapan dari bahasa Inggris yaitu *effective* yang artinya berhasil. Menurut kamus ilmiah popular, efektivitas diartikan sebagai hal yang berpengaruh atau keberhasilan.

Definisi efektivitas juga disampaikan oleh Mustika Rihardini (2012) yang menyebutkan efektivitas sebagai suatu ukuran yang menyatakan seberapa jauh target yang telah dicapai oleh manajemen, yang mana target tersebut sudah ditentukan terlebih dahulu.

Tingkat efektivitas dapat diukur dengan membandingkan antara rencana yang telah dibuat dengan hasil nyata yang diperoleh. Jika hasil nyata yang diperoleh tidak mencapai tujuan atau sasaran yang diharapkan maka dikatakan hal tersebut kurang efektif.

Pada skripsi ini, yang dijadikan sebagai rencana adalah selisih nilai perhitungan dari metode pendekatan dengan perhitungan *software* WinQSB 2.0 kurang dari 0,1 %, sehingga jika selisih tersebut lebih dari 0,1 % maka dikatakan tujuan tidak tercapai, atau metode pendekatan tersebut kurang efektif untuk digunakan.

B. Optimasi

Optimasi merupakan salah satu cabang ilmu matematika yang fokus digunakan untuk mendapatkan nilai minimum atau maksimum secara sistematis dari suatu fungsi maupun pencarian nilai lainnya dalam berbagai kasus (Qoriatun Maryamah, 2013 : 13).

Definisi lain yaitu menurut Licker (2003 : 170), optimasi yang berasal dari kata bahasa Inggris *optimization* memiliki arti memaksimumkan atau meminimumkan sebuah fungsi yang diberikan untuk beberapa macam kendala.

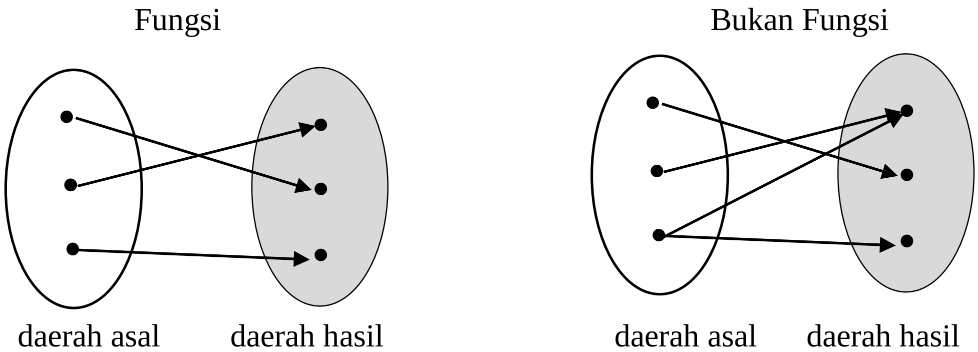
Berdasarkan beberapa definisi tersebut, maka disimpulkan bahwa optimasi adalah suatu cabang ilmu dalam matematika untuk memaksimumkan atau meminimumkan fungsi tujuan dengan mempertimbangkan beberapa kendala yang diberikan.

C. Fungsi

Definisi 2.1. Fungsi (Varberg & Purcell, 2001 : 57)

Suatu fungsi f adalah suatu aturan padanan yang menghubungkan setiap obyek x dalam suatu himpunan, yang disebut daerah asal, dengan sebuah nilai tunggal f(x) dari suatu himpunan kedua. Himpunan nilai yang diperoleh secara demikian disebut daerah hasil fungsi.

Gambar 2.1 berikut memberikan ilustrasi untuk membedakan suatu fungsi dengan bukan fungsi.



Gambar 2. 1 Ilustrasi Fungsi dan Bukan Fungsi

Fungsi yang terbentuk dari suatu konstanta disebut fungsi konstanta, sedangkan fungsi yang terbentuk dari suatu variabel maka disebut fungsi identitas. Fungsi yang diperoleh dari fungsi konstanta dan fungsi identitas dengan menggunakan operasi penambahan, pengurangan, dan perkalian disebut fungsi polinom. (Varberg & Purcell, 2001 : 71)

Suatu fungsi polinom dapat ditulis dalam bentuk $f(x) = a_n x^n + a_{n-1}x^{n-1} + \cdots + a_1 x + a_0$ dengan koefisien-koefisien a berupa bilangan real dan n adalah bilangan bulat tak negatif. Secara khusus, bentuk $f(x) = ax + b$ merupakan fungsi polinom derajat satu dan disebut fungsi linear (Varberg & Purcell, 2001 : 71).

Selain bentuk fungsi linear, terdapat juga bentuk fungsi nonlinear. Fungsi nonlinear yang terbentuk dari fungsi polinom derajat dua disebut juga sebagai fungsi kuadrat. Pada fungsi kuadrat dikenal konsep kecembungan (*convexity*). Konsep ini akan dijelaskan dalam Definisi 2.2 berikut.

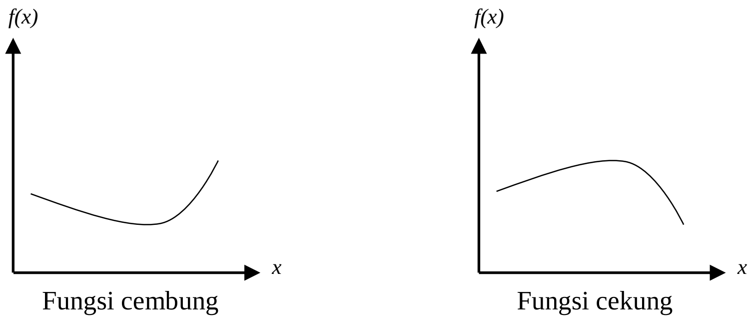
Definisi 2.2. Fungsi Cembung (Hillier & Lieberman, 2001 : 1159)

Fungsi satu variabel f(x) adalah fungsi cembung jika untuk setiap pasangan nilai x, misalnya x' dan x" berlaku

$$f(\lambda x'' + (1 - \lambda)x') \leq \lambda f(x'') + (1 - \lambda) f(x')$$

untuk seluruh nilai λ dengan $0 \leq \lambda \leq 1$. Fungsi ini disebut ***fungsi cembung sempurna*** *(strictly convex function) jika ≤ (kurang dari sama dengan) dapat diganti dengan < (kurang dari). Fungsi ini disebut* ***fungsi cekung*** *jika pernyataan ≤ dapat diganti oleh ≥ (lebih dari sama dengan) atau* ***fungsi cekung sempurna*** *(strictly concave function) jika pernyataan ≤ dapat diganti oleh > (lebih dari).*

Gambar 2.2 berikut merupakan ilustrasi dari bentuk kurva fungsi cembung dan fungsi cekung.



Gambar 2. 2 Ilustrasi bentuk fungsi cembung dan cekung

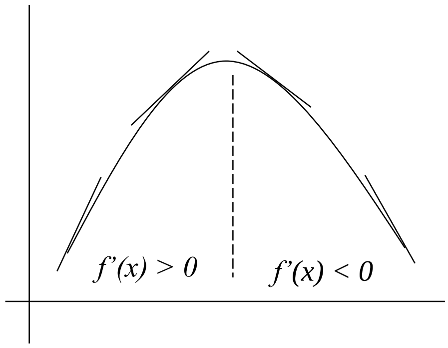
Selain dengan menggunakan Definisi 2, fungsi cekung dan fungsi cembung juga dapat ditentukan dengan menggunakan turunan kedua. Menurut Hillier & Lieberman (2001), fungsi cekung dan cembung pada suatu fungsi satu variabel dapat ditentukan melalui Teorema 2.2.1 berikut :

Teorema 2.2.1. Tes Kecembungan fungsi satu variabel (Hillier & Lieberman, 2001 : 1159)

1. *f(x) cembung jika dan hanya jika turunan kedua f(x) yaitu* $\frac{d^2 f(x)}{dx^2} \geq 0$ *untuk setiap nilai x yang diberikan.*
2. *f(x) cembung sempurna jika dan hanya jika turunan kedua f(x) yaitu* $\frac{d^2 f(x)}{dx^2} > 0$ *untuk setiap nilai x yang diberikan.*
3. *f(x) cekung jika dan hanya jika turunan kedua f(x) yaitu* $\frac{d^2 f(x)}{dx^2} \leq 0$ *untuk setiap nilai x yang diberikan.*

4. *f(x) cekung sempurna jika dan hanya jika turunan kedua f(x) yaitu* $\frac{d^2f(x)}{dx^2} < 0$ *untuk setiap nilai x yang diberikan.*

   Bukti :



Gambar 2. 3 Gradien kurva *f(x)*

Secara geometri, suatu nilai *f(x)* yang kontinu memiliki garis-garis singgung yang dinyatakan sebagai $\frac{df(x)}{dx}$ atau *f'(x)* seperti tampak pada Gambar 2.3. Jika nilai $\frac{df(x)}{dx} > 0$ pada selang *I* maka fungsi naik pada selang *I*, begitu juga sebaliknya. Apabila garis singgung berbelok searah jarum jam (seperti Gambar 2.3) yaitu saat turunan kedua *f(x)* yaitu $\frac{d^2f(x)}{dx^2}$ positif maka *f(x)* merupakan fungsi cekung, namun jika berbelok berlawanan arah jarum jam yaitu saat turunan kedua *f(x)* yaitu $\frac{d^2f(x)}{dx^2}$ negatif maka *f(x)* berupa fungsi cembung.

D. Pemrograman Linear

Pemrograman linear adalah teknik pengambilan keputusan untuk memecahkan masalah pengalokasian sumber daya untuk berbagai kepentingan seoptimal mungkin. Teknik ini dikembangkan oleh LV Kantorovich pada tahun 1939 dan merupakan salah satu metode dalam riset operasi (Eddy Herjanto, 2007 : 43).

Definisi lain pemrograman linear juga disampaikan oleh Siswanto (2007 : 26) yang menyebutkannya sebagai metode matematis yang berkarakteristik linear untuk menemukan suatu penyelesaian optimal dengan cara memaksimumkan atau meminimumkan fungsi tujuan terhadap satu susunan kendala.

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

1. Variabel keputusan.

   Variabel keputusan adalah variabel yang akan 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.

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

Secara umum, masalah program linear dapat dirumuskan menjadi bentuk berikut :

Memaksimumkan / meminimumkan : $f = C^T X$ (2.1) dengan kendala : $AX = B$, dan $X \geq 0$ (2.2)

Dalam hal ini, $C^T = [c_1 \quad c_2 \quad \cdots \quad c_n]$, (2.3)

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \tag{2.4}$$

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}, \tag{2.5}$$

dan $$B = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix} \tag{2.6}$$

Matriks X merupakan matriks satu kolom dari variabel-variabel yang dicari, dan C^T adalah matriks satu baris untuk setiap koefisien ongkos (c_j). Matriks A merupakan matriks koefisien persamaan kendala, dan B adalah matriks satu kolom dari ruas kanan persamaan kendala. (Bronson & Naadimuthu, 1997 : 20)

Jika (2.1) dan (2.2) dituliskan semua dalam bentuk matriks maka akan menjadi :

Memaksimumkan atau meminimumkan $f = [c_1 \quad c_2 \quad \cdots \quad c_n] \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$, dengan kendala :

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix} \text{ , dan } \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \geq 0.$$

Jika bentuk perkalian matriks tersebut diuraikan menjadi penjumlahan aljabar akan menjadi :

Memaksimumkan atau meminimumkan

$$f = c_1x_1 + c_2x_2 + \cdots + c_nx_n = \sum_{j=1}^{n} c_jx_j \qquad (2.7)$$

dengan kendala :

$$a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \qquad (2.8a)$$

$$a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \qquad (2.8b)$$

$$\vdots$$

$$a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n = b_m \qquad (2.8c)$$

$$x_1, x_2, \dots, x_n \geq 0 \qquad (2.8d)$$

atau jika ditulis ulang, maka bentuk fungsi kendala (2.8a) – (2.8d) menjadi :

$$g_i(x_1, x_2, \dots, x_n) = \sum_{j=1}^{n} a_{ij}x_j = b_i \text{ dengan } i = 1,2, \dots, m \qquad (2.9a)$$

$$x_j \geq 0 \text{ , dengan } j = 1,2, \dots, n \qquad (2.9b)$$

E. Metode Simpleks

Masalah optimasi untuk program linear dengan dua variabel diselesaikan dengan menggunakan metode penggambaran grafik. Pada masalah optimasi dengan fungsi tujuan yang memiliki lebih dari tiga variabel lebih, metode grafik sudah tidak memungkinkan digunakan sehingga digunakan cara lain yaitu metode simpleks. Kendala dari permasalahan yang diselesaikan dengan metode simpleks harus diubah dulu ke dalam bentuk kanonik. Bentuk kanonik yaitu kondisi dimana semua kendala berbentuk persamaan linear (B. Susanta, 1994 : 70).

Menurut Eddy Herjanto (2007 : 45), cara mengubah suatu kendala menjadi bentuk kanonik adalah sebagai berikut :

1. Menambahkan variabel *slack* (s) untuk kendala yang berbentuk pertidaksamaan kurang dari sama dengan (≤).
2. Mengurangi dengan variabel *surplus* (e) untuk kendala yang berbentuk pertidaksamaan lebih dari sama dengan (≥).
3. Mengalikan dengan -1 terhadap nilai suku tetap (b_i) negatif.

Variabel *slack* maupun variabel *surplus* merupakan variabel yang membuat ruas yang semula tak seimbang menjadi seimbang, sehingga antara ruas kiri dan ruas kanan bernilai sama (B. Susanta, 1994 : 69).

Mengingat variabel *surplus* tidak bisa menjadi basis karena koefisiennya negatif maka perlu ditambahkan suatu variabel yang bernilai positif untuk menjadi basis, variabel tersebut dinyatakan sebagai variabel buatan (a) (B. Susanta, 1994 : 88).

Pada tabel optimal, karena berfungsi sebagai penyeimbang maka variabel *surplus* harus bernilai nol. Agar variabel *surplus* segera bernilai nol maka disusunlah fungsi sasaran baru dengan bentuk $\bar{f} = f - Ma$, dengan f adalah fungsi tujuan awal, a adalah suatu variabel buatan, dan M merupakan bilangan positif yang cukup besar. Hal ini diharapkan supaya a segera keluar dari basis karena koefisien ongkosnya negatif besar. (B. Susanta, 1994 : 89)

Setelah bentuk kanonik dari setiap kendala sudah didapatkan, maka langkah selanjutnya adalah membuat tabel simpleks.

Tabel 2. 1 Bentuk tabel simpleks

	c_j	c_1	c_2	...	c_n		
C_i	X_i/x_j	x_1	x_2	...	x_n	b_i	R_i
C_1	X_1	a_{11}	a_{12}	...	a_{1n}	b_1	R_1
C_2 $\vdots$	X_2 $\vdots$	a_{21} $\vdots$	a_{22}	...	a_{2n}	b_2 $\vdots$	R_2 $\vdots$
C_m	X_m	a_{m1}	a_{m2}	...	a_{mn}	b_m	R_m
	Z_j	Z_1	Z_2	...	Z_n	Z_t	
	$Z_j - c_j$	$Z_1 - c_1$	$Z_2 - c_2$	...	$Z_n - c_n$		

Keterangan :

x_j : variabel-variabel yang akan dicari

X_i : variabel yang menjadi basis dalam tabel yang ditinjau

c_j : koefisien ongkos

C_i : koefisien ongkos milik variabel basis X_i

a_{ij} : koefisien dalam kendala utama

b_i : suku tetap (nilai di ruas kanan fungsi kendala)

Z_j : $\sum_{i=1}^{m} C_i a_{ij}$ (jumlah hasil kali C_i dengan kolom a_{ij})

Z_t : $\sum_{i=1}^{m} C_i b_i$ (jumlah hasil kali C_i dengan kolom b_i)

$Z_j - c_j$: selisih Z_j dengan c_j

Apabila tabel yang bersangkutan belum optimal dan dipilih x_k sebagai basis baru maka disusun kolom R_i yang diperoleh dengan :

$R_i = \frac{b_i}{a_{ik}}$, hanya untuk $a_{ik} > 0$ (B. Susanta, 1994 : 74)

Kasus dimana semua fungsi kendalanya berupa pertidaksamaan satu jenis disebut sebagai kasus maksimum atau minimum baku.

Pada kasus memaksimumkan, tabel simpleks dinyatakan telah mencapai optimal jika $Z_j - c_j \geq 0$ untuk semua nilai j. Jika tabel belum optimal maka dilakukan perbaikan tabel (iterasi). Pada kasus memaksimumkan, x_k terpilih adalah yang memiliki $Z_k - c_k < 0$ yang paling kecil, karena jika diambil $Z_k - c_k > 0$ maka nilai fungsi tujuan akan menjauhi nilai optimal. Variabel yang terpilih menjadi basis baru adalah variabel yang memiliki nilai R_i terkecil. Sebaliknya untuk kasus meminimumkan, x_k terpilih adalah yang memiliki $Z_k - c_k > 0$ yang paling besar. Variabel yang menjadi basis baru pada tabel perbaikan adalah variabel yang memiliki nilai R_i terkecil, dengan tabel optimumnya dicapai saat $Z_j - c_j \leq 0$ untuk semua nilai j (B. Susanta, 1994 : 77-78).

Contoh 2.1 berikut ini digunakan sebagai ilustrasi untuk menambah penjelasan terkait metode simpleks.

Contoh 2.1 :

Diketahui suatu permasalahan program linear sebagai berikut (B. Susanta, 1994 : 88) :

$x_1 + x_2 + 2x_3 \ \le 12$ (2.10a)

$2x_1 - 6x_2 - x_3 \ \ge 4$ (2.10b)

dan memaksimumkan $f = -8x_1 + 6x_2 + 8x_3$ (2.11)

Akan diselesaiakan masalah program linear tersebut dengan menggunakan metode simpleks.

Langkah pertama yang dilakukan yaitu menyusun bentuk kanonik dari (2.10) dan (2.11), yaitu :

$x_1 + x_2 + 2x_3 + s = 20$ (2.12a)

$2x_1 - 6x_2 - x_3 - e + a = \ 50$ (2.12b)

Serta memaksimumkan $\bar{f} = -8x_1 + 6x_2 + 8x_3 + 0s + 0e - Ma$

Variabel *slack* untuk persamaan (2.10a) dan (2.10b) adalah s, sedangkan variabel *surplus*nya yaitu *e*, dan *a* sebagai variabel buatan.

Setelah didapatkan bentuk kanonik, selanjutnya diselesaikan dengan menggunakan metode simpleks.

Tabel 2. 2 Bentuk tabel simpleks contoh 2.1 (iterasi 1)

	c_j	-8	6	8	0	0	-*M*		
C_i	X_i/x_j	x_1	x_2	x_3	s	e	a	b_i	R_i
0	s	1	1	2	1	0	0	12	12
-*M*	a	2	-6	-1	0	-1	1	4	2
	Z_j	-2*M*	6*M*	*M*	0	0	-*M*	-4*M*	
	$Z_j - C_j$	-2*M*+8	6*M*-6	*M*-8	0	0	0		

Pada Tabel 2.2 diketahui nilai $Z_j - c_j$ terkecil yaitu untuk variabel x_1, dengan nilai R_i terkecil yang tidak negatif adalah pada variabel *a*, sehingga variabel x_1 menjadi basis baru menggantikan variabel *a*.

Tabel 2. 3 Proses Iterasi 2 contoh 2.1

	c_j	-8	6	8	0	0	-M		
C_i	X_i/x_j	x_1	x_2	x_3	s	e	a	b_i	R_i
0	s	0	4	2,5	1	0,5	-0,5	10	4
-8	x_1	1	-3	-0,5	0	-0,5	0,5	2	-4
	Z_j	-8	24	4	0	4	-4	-16	
	$Z_j - C_j$	0	18	-4	0	0	0		

Pada Tabel 2.3 diketahui nilai $Z_j - c_j$ terkecil yaitu pada variabel x_3, dengan nilai R_i terkecil yang tidak negatif adalah pada variabel s, sehingga variabel x_3 menjadi basis baru menggantikan variabel s.

Tabel 2. 4 Proses Iterasi 3 contoh 2.1

	C_j	-8	6	8	0	0	-M		
c_i	x_i/X_j	x_1	x_2	x_3	s	e	a	b_i	R_i
8	x_3	0	1,6	1	0,4	0,2	-0,2	4	
-8	x_1	1	-2,2	0	0,2	-0,4	0,4	4	
	Z_j	-8	30,4	8	1,6	4,8	-4,8	0	
	$Z_j - C_j$	0	24,4	0	1,6	4,8	M-4,8		

Pada Tabel 2.4, semua nilai $Z_j - c_j \geq 0$ sehingga tabel telah optimal. Solusi dari permasalahan Contoh 2.1 yaitu nilai f maksimal = 0 dengan nilai variabel x_3 = 4, x_1 = 4, dan x_2 = 0.

F. Teorema Dualitas

Konsep dualitas menyebutkan bahwa suatu masalah pemrograman linear berkaitan dengan masalah pemrograman linear yang lain, dalam hal ini disebut sebagai dual.

Secara umum, bentuk masalah primal dan dual dapat dituliskan sebagai berikut :

<u>Primal</u>

Memaksimumkan / meminimumkan : $f = C^T X$ (2.13a)

dengan kendala : $AX = B,$ dan $X \geq 0$ (2.13b)

<u>Dual</u>

Meminimumkan / memaksimumkan : $f = B^T Y$ (2.14a)

dengan kendala : $A^T Y\ (\geq, \leq) C$ (2.14b)

Dalam hal ini, $B^T = [b_1 \quad b_2 \quad \dots \quad b_m],$ (2.15)

$$A^T = \begin{bmatrix} a_{11} & a_{21} & \cdots & a_{m1} \\ a_{12} & a_{22} & \cdots & a_{m2} \\ \vdots & \vdots & & \vdots \\ a_{1n} & a_{2n} & \cdots & a_{mn} \end{bmatrix}, \tag{2.16}$$

$$C = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix}, \tag{2.17}$$

$$\text{dan } Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}. \tag{2.18}$$

Matriks A^T dan B^T merupakan transpose dari matriks A dan matriks B, sedangkan C adalah matriks satu kolom untuk setiap koefisien ongkos (c_j), dan Y merupakan matriks satu kolom dari variabel-variabel dual yang dicari (Bronson & Naadimuthu, 1997 : 56).

Jika (2.14a) dan (2.14b) langsung ditulis dalam bentuk matriks secara keseluruhan, maka akan didapat bentuk :

Meminimumkan / memaksimumkan : $f = \begin{bmatrix} b_1 & b_2 & \cdots & b_m \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}$

dengan kendala : $\begin{bmatrix} a_{11} & a_{21} & \cdots & a_{m1} \\ a_{12} & a_{22} & \cdots & a_{m2} \\ \vdots & \vdots & & \vdots \\ a_{1n} & a_{2n} & \cdots & a_{mn} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} (\geq, \leq) \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix}$.

Bentuk perkalian matriks tersebut jika diuraikan menjadi penjumlahan aljabar akan menjadi :

Meminimumkan atau memaksimumkan

$$f = b_1 y_1 + b_2 y_2 + \cdots + b_m y_m = \sum_{i=1}^{m} b_i y_i \tag{2.19}$$

dengan kendala :

$$a_{11} y_1 + a_{21} y_2 + \cdots + a_{m1} y_m \ (\geq, \leq)\ c_1 \tag{2.20a}$$

$$a_{12} y_1 + a_{22} y_2 + \cdots + a_{m2} y_m \ (\geq, \leq)\ c_2 \tag{2.20b}$$

$\vdots$

$$a_{1n} y_1 + a_{2n} y_2 + \cdots + a_{mn} y_m \ (\geq, \leq)\ c_n \tag{2.20c}$$

Lemma 2.1. Dualitas Lemah (Winston, 2003 :)

Jika $\bar{X} = \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \\ \vdots \\ \bar{x}_n \end{bmatrix}$ *merupakan solusi layak masalah primal, dan* $\bar{Y} = \begin{bmatrix} \bar{y}_1 \\ \bar{y}_2 \\ \vdots \\ \bar{y}_m \end{bmatrix}$ *merupakan solusi layak masalah dual, maka nilai f primal ≤ f dual.*

Bukti :

Pada permasalahan primal yang dinyatakan dalam bentuk

Memaksimumkan $f = c_1 x_1 + c_2 x_2 + \cdots + c_n x_n = \sum_{j=1}^{n} c_j x_j$

dengan kendala : $a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n \leq b_i$ dengan i=1, 2, …, m.

Bentuk masalah dual menjadi

Meminimumkan $f = b_1y_1 + b_2y_2 + \cdots + b_my_m = \sum_{i=1}^{m} b_iy_i$

dengan kendala : $a_{1j}y_1 + a_{2j}y_2 + \cdots + a_{mj}y_m \geq c_n$ dengan j=1, 2, …, n.

Diperhatikan bahwa $y_i \geq 0$, jika dikalikan dengan kendala masalah primal maka menjadi $y_ia_{i1}x_1 + y_ia_{i2}x_2 + \cdots + y_ia_{in}x_n \leq b_iy_i$ untuk i=1, 2, …, m. Jika terdapat sejumlah m kendala, maka

$$\sum_{i=1}^{m}\sum_{j=1}^{n} y_ia_{ij}x_j \leq \sum_{i=1}^{m} b_iy_i \quad (2.21a)$$

Diperhatikan bahwa $x_j \geq 0$, jika dikalikan dengan kendala masalah dual maka menjadi $x_ja_{1j}y_1 + x_ja_{2j}y_2 + \cdots + x_ja_{mj}y_m \geq c_jx_j$ untuk j=1, 2, …, n. Jika terdapat sejumlah n variabel, maka

$$\sum_{i=1}^{m}\sum_{j=1}^{n} y_ia_{ij}x_j \geq \sum_{j=1}^{n} c_jx_j \quad (2.21b)$$

Berdasarkan (2.21a) dan (2.21b) maka didapatkan

$\sum_{j=1}^{n} c_jx_j \leq \sum_{i=1}^{m}\sum_{j=1}^{n} y_ia_{ij}x_j \leq \sum_{i=1}^{m} b_iy_i$, atau $\sum_{j=1}^{n} c_jx_j \leq \sum_{i=1}^{m} b_iy_i$.

Terbukti bahwa nilai f primal ≤ f dual.

Teorema 2.2. Teorema Dualitas (Bronson & Naadimuthu, 1997 : 56)

Jika terdapat sebuah pemecahan optimal bagi suatu program primal atau dual simetris, maka program lainnya juga memiliki suatu pemecahan optimal dan kedua fungsi tujuannya memiliki nilai optimal yang sama.

Bukti :

Berdasarkan Lemma 2.1, maka $C^TX \leq B^T\bar{Y}$. Suatu titik layak pada masalah primal harus menghasilkan sebuah nilai f primal yang tidak melebihi $B^T\bar{Y}$.

Mengingat $\bar{X}$ adalah solusi layak primal dan punya suatu nilai fungsi tujuan primal yang memenuhi $C^T\bar{X} = B^T\bar{Y}$, maka $\bar{X}$ haruslah solusi optimal primal. Hal yang serupa, karena $\bar{X}$ solusi layak primal, dual lemah mengisyaratkan bahwa untuk suatu titik layak dual Y, maka $C^T\bar{X} \leq B^TY$. Suatu titik layak dual harus menghasilkan sebuah nilai fungsi tujuan yang melebihi $C^T\bar{X}$. Mengingat $\bar{Y}$ merupakan solusi layak dual dan punya sebuah nilai fungsi tujuan dual yang memenuhi $B^T\bar{Y} = C^T\bar{X}$, maka $\bar{Y}$ haruslah solusi optimal dual.

Berdasarkan penjelasan tersebut maka Teorema 2.2 terbukti.

Suatu pemrograman linear dikatakan berbentuk simetris jika semua variabel dibatasi bernilai non negatif dan semua kendala berupa pertidaksamaan. Pada kasus memaksimumkan, semua kendala berupa pertidaksamaan kurang dari sama dengan (≤), sedangkan kasus minimasi memiliki kendala dengan pertidaksamaan lebih dari sama dengan (≥) (Sri Mulyono, 1991 : 62).

Jika suatu permasalahan belum memenuhi kondisi simetris maka dapat diubah menjadi simetris. Adapun cara mengubah bentuk tak simetris menjadi simetris menurut Sri Mulyono (1991 : 66) yaitu :

1. Kendala pertidaksamaan ≥ pada masalah memaksimumkan dikalikan (-1) sehingga tanda berubah menjadi ≤.
2. Kendala persamaan (=) dapat diubah menjadi dua kendala yaitu pertidaksamaan ≥ dan pertidaksamaan ≤.

Berdasarkan Teorema 2.2, dualitas dapat digunakan untuk memeriksa kembali tabel optimal pada masalah primal (Pangestu Subagyo, dkk., 2000 : 62). Menurut Hamdy A. Taha (1999 : 151) pemecahan optimal untuk kedua masalah (primal dan dual) didapatkan jika nilai tujuan keduanya sama .

Contoh 2.2 berikut akan menjelaskan penerapan dualitas dalam suatu permasalahan.

Contoh 2.2 :

Suatu pabrik A memproduksi dua jenis barang yaitu x_1 dan x_2. Baik barang x_1 maupun x_2 membutuhkan 3 buah komponen dalam pembuatannya (n_1, n_2, n_3) dengan kadar yang berbeda dan dinyatakan sebagai a_{ij}. Persediaan maksimal komponen yang tersedia tiap minggu dinyatakan sebagai b_j , sedangkan keuntungan yang diperoleh pabrik A dinyatakan sebagai c_i.

Berdasarkan ilustrasi Contoh 2.2, masalah program linear dapat direpresentasikan dalam Tabel 2.5.

Tabel 2. 5 Tabel dualitas Contoh 2.2

Komponen	Jenis 1 (x_1)	Jenis 2 (x_2)	Batas maksimal
n_1	a_{11}	a_{12}	b_1
n_2	a_{21}	a_{22}	b_2
n_3	a_{31}	a_{33}	b_3
Batas minimal	c_1	c_2	

Pada Tabel 2.5, jika dibaca ke bawah maka akan menjadi masalah dual. Sedangkan jika dibaca ke kanan maka didapatkan masalah primal. Maka hasil dari pembacaan tabel tersebut yaitu :

Masalah dual :

Memaksimumkan $f = c_1x_1 + c_2x_2$

dengan kendala :

$a_{11}x_1 + a_{12}x_2 \leq b_1$

$a_{21}x_1 + a_{22}x_2 \leq b_2$

$a_{31}x_1 + a_{32}x_2 \leq b_3$

Sedangkan masalah primalnya menjadi

Meminimumkan $g = b_1n_1 + b_2n_2 + b_3n_3$

dengan kendala :

$a_{11}n_1 + a_{21}n_2 + a_{31}n_3 \geq c_1$

$a_{12}n_1 + a_{22}n_2 + a_{33}n_3 \geq c_2$

Keoptimalan masalah dual dan masalah primal mengakibatkan suatu kondisi yang disebut dengan kondisi *complementary slackness* (B. Susanta, 1994 : 186) :

1. Jika dalam penyelesaian optimal masalah primal, kendala ke-*h* berupa pertidaksamaan maka dalam penyelesaian optimal masalah dual variabel ke-*h* bernilai nol.
2. Jika dalam penyelesaian optimal masalah primal, variabel ke-*p* bernilai positif (kendala tak negatif untuk x_p berupa pertidaksamaan $x_p > 0$) maka dalam penyelesaian optimal masalah dual kendala ke-*p* akan berupa persamaan.

Pada kondisi *complementary slackness* tersebut dapat ditulis secara matematis yaitu :

1. $s_hy_h = 0$
2. $x_pe_p = 0$

G. Pemrograman Nonlinear

Pada penerapan pemrograman linear, asumsi penting yang harus dipenuhi adalah bahwa semua fungsi berupa linear. Sering kali dalam permasalahan nyata sehari-hari asumsi penting ini tidak dapat terpenuhi. Hal inilah yang kemudian melahirkan konsep baru yaitu masalah pemrograman nonlinear. Menurut Hiller & Lieberman (2001 : 654) bentuk umum dari pemrograman nonlinear adalah menemukan nilai $\mathbf{x} = (x_1, x_2, \dots, x_n)$ sehingga

Meminimumkan / memaksimumkan

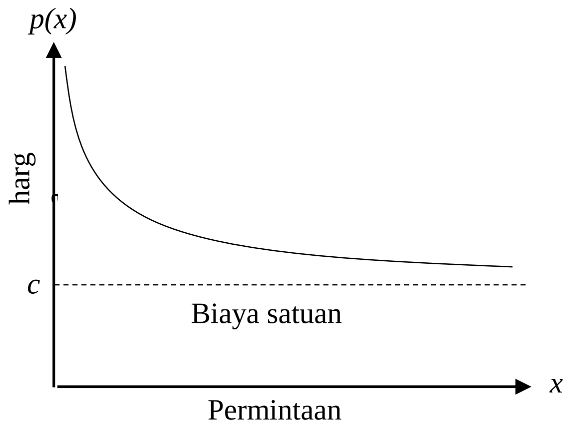
$f(\mathbf{x})$, dimana $f(\mathbf{x})$ berupa fungsi - fungsi nonlinear, (2.22)

dengan kendala $g_i(\mathbf{x}) \leq b_i$, untuk setiap $i = 1,2, \dots, m$ (2.23a)

dan $x \geq 0$. (2.23b)

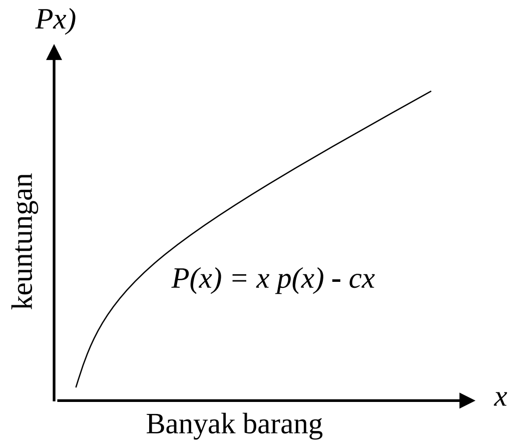
Fungsi kendala $g_i(\mathbf{x})$ dapat berupa fungsi nonlinear ataupun fungsi linear. Selain itu, $f(\mathbf{x})$ dan fungsi $g_i(\mathbf{x})$ adalah fungsi – fungsi dengan n variabel.

Salah satu contoh penggunaan pemrograman nonlinear adalah tentang masalah produk campuran dan elastisitas harga. Suatu perusahaan besar memiliki kemungkinan menghadapi elastisitas harga dimana banyaknya barang yang dijual berbanding terbalik dengan harganya. Gambar 2.4 berikut menjelaskan kurva harga permintaan produk pada perusahaan.



Gambar 2. 4 Kurva Harga Permintaan

Nilai dari *p(x)* adalah harga yang ditetapkan agar terjual *x* satuan barang. Jika biaya satuan produksi barang selalu konstan (c), maka keuntungan perusahaan akan dinyatakan dalam bentuk fungsi nonlinear $P(x) = x\, p(x) - cx$ seperti yang dijelaskan dalam Gambar 2.5



Gambar 2. 5 Fungsi Keuntungan

Apabila terdapat *n* jenis produk yang dihasilkan dengan fungsi keuntungan serupa, dimana $P_j(x_j)$ menyatakan fungsi keuntungan dari penjualan x_j satuan dari produk *j* (*j* = 1,2,…*n*), maka fungsi tujuan secara keseluruhan adalah $f(\mathbf{x}) =$

$\sum_{j=1}^{n} P_j(x_j)$, yaitu penjumlahan dari beberapa fungsi keuntungan yang nonlinear. (Hillier & Lieberman, 2001 : 655 - 656)

H. Kondisi Karush Kuhn-Tucker (KKT *conditions*)

Metode Karush Kuhn Tucker dapat dipergunakan untuk mencari solusi yang optimal dari suatu fungsi linear maupun nonlinear. Pada metode Karush Kuhn Tucker, program yang diselesaikan berupa program yang memiliki kendala pertidaksamaan. Metode Karush Kuhn Tucker merupakan pengembangan dari penyelesaian model nonlinear berkendala persamaan yang dikerjakan dengan mencari titik – titik stasionernya, yaitu titik yang berpotensi menjadi titik optimal.

Terdapat beberapa syarat Karush Kuhn Tucker untuk masalah optimasi berkendala. Syarat tersebut dirumuskan oleh Karush dan Kuhn Tucker. Teorema 2.3 dan 4 merupakan syarat KKT untuk masalah maksimasi dan minimasi.

Teorema 2.3. Syarat KKT masalah maksimasi (Winston, 2003 : 676)

Misalkan $f(\mathbf{x})$ *dan* $g_i(\mathbf{x})$ *merupakan suatu masalah berpola memaksimumkan. Jika* $\mathbf{x} = (x_1, x_2, \dots, x_n)$ *merupakan suatu solusi optimal untuk* $f(\mathbf{x})$ *dan* $g_i(\mathbf{x})$, *maka* $\mathbf{x} = (x_1, x_2, \dots, x_n)$ *harus memenuhi (2.22) dan terdapat pengali* $\lambda_1, \lambda_2, \dots, \lambda_m$ *serta variabel slack* $s_1, s_2, \dots, s_n$ *sehingga memenuhi*

1. $\frac{\partial f}{\partial x_j} - \sum_{i=1}^{m} \lambda_i \frac{\partial g_i}{\partial x_j} + s_j = 0,$ *untuk j = 1,2, …, n*
2. $\lambda_i [\, b_i - g_i(\mathbf{x})] = 0,$ *untuk i = 1,2, …, m*
3. $(\frac{\partial f}{\partial x_j} - \sum_{i=1}^{m} \lambda_i \frac{\partial g_i}{\partial x_j}) x_j = 0,$ *untuk j = 1,2, …, n*
4. $\lambda_i \geq 0$, *untuk i = 1,2, …, m*
5. $s_j \geq 0,$ *untuk j = 1,2, …, n*

Teorema 2.4. Syarat KKT masalah minimasi (Winston, 2003 : 676)

Misalkan $f(\mathbf{x})$ *dan* $g_i(\mathbf{x})$ *merupakan suatu masalah berpola meminimumkan. Jika* $\mathbf{x} = (x_1, x_2, \dots, x_n)$ *merupakan suatu solusi optimal untuk* $f(\mathbf{x})$ *dan* $g_i(\mathbf{x})$, *maka* $\mathbf{x} = (x_1, x_2, \dots, x_n)$ *harus memenuhi (2.22) dan terdapat pengali* $\lambda_1, \lambda_2, \dots, \lambda_m$ *serta variabel surplus* $e_1, e_2, \dots, e_n$ *sehingga memenuhi*

1. $\frac{\partial f}{\partial x_j} + \sum_{i=1}^{m} \lambda_i \frac{\partial g_i}{\partial x_j} - e_j = 0,$ *untuk j = 1,2, …, n*
2. $\lambda_i [\, b_i - g_i(\mathbf{x})] = 0,$ *untuk i = 1,2, …, m*
3. $(\frac{\partial f}{\partial x_j} + \sum_{i=1}^{m} \lambda_i \frac{\partial g_i}{\partial x_j}) x_j = 0,$ *untuk j = 1,2, …, n*
4. $\lambda_i \geq 0$, *untuk i = 1,2, …, m*
5. $e_j \geq 0,$ *untuk j = 1,2, …, n*

Pada syarat kedua dari Teorema 2.3 dan Teorema 2.4 berakibat $g_i(\mathbf{x}) - b_i \leq 0$. Hal ini dapat dilihat pada saat $\lambda_i = 0$, sehingga $[\, b_i - g_i(\mathbf{x})] \neq 0$. Berdasarkan bentuk umum fungsi kendala, maka $[\, b_i - g_i(\mathbf{x})] > 0$ yaitu $g_i(\mathbf{x}) \leq b_i$.

I. *Quadratic Programming*

1. Bentuk Umum Model Nonlinear untuk *Quadratic Porgramming*

Quadratic Programming merupakan pendekatan penyelesaian permasalahan optimasi nonlinear dimana kendalanya berupa fungsi linear dan fungsi tujuannya merupakan kuadrat dari variabel keputusan ataupun perkalian dari dua variabel keputusan (Hillier & Lieberman, 2001 : 665).

Model nonlinear yang akan diselesaikan dengan menggunakan *quadratic programming* memiliki bentuk umum yaitu (Peressini, et al., 1988 : 258) :

Meminimumkan $f(X) = C^T X + \frac{1}{2} X^T Q X + d$ (2.24a)

dengan kendala : $AX \leq B,\ X \geq 0$ (2.24b)

Konsep matriks $C^T, X, A,$ dan B masih sama seperti yang telah dijelaskan pada (2.3) – (2.6). Adapun *d* merupakan suatu konstanta, sedangkan Q merupakan matriks simetris yang tersusun dari nilai q_{ij}, dimana q_{ij} merupakan hasil dari turunan parsial kedua terhadap x_i dan x_j dari fungsi tujuan. Matriks Q merupakan matriks simetris, sehingga nilai $q_{ij} = q_{ji}$. Bentuk (2.24a) ini juga dapat ditransformasikan menjadi bentuk penjumlahan aljabar biasa, yaitu :

$$f(X) = C^T X + \frac{1}{2} X^T Q X + d = \sum_{j=1}^{n} c_j x_j + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} q_{ij} x_i x_j + d \qquad (2.25)$$

Sebagai penjelasan terkait bentuk umum model nonlinear untuk *quadratic programming*, akan diberikan ilustrasi dalam Contoh 2.3 berikut.

Contoh 2.3 :

Suatu masalah program nonlinear berikut akan diidentifikasi apakah merupakan bentuk permasalahan *quadratic programming* atau tidak, yaitu :

Meminimumkan $f(x_1, x_2) = 15\, x_1 + 30\, x_2 + 4\, x_1 x_2 - 2\, {x_1}^2 - 4{x_2}^2$, dengan kendala $x_1 + 2x_2 \leq 30$ dan $x_1, x_2 > 0$.

Pada masalah ini, bentuk model nonlinear yang diselesaikan dengan *quadratic programming* dapat dicari yaitu dengan menentukan C^T, X, A, Q dan b.

Matriks C^T merupakan matriks baris koefisien – koefisien dari *x* sehingga $C^T = [15 \quad 30]$.

Matriks X adalah matriks kolom untuk variabel-variabel keputusan, sehingga $X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, sedangkan $Q = \begin{bmatrix} -4 & 4 \\ 4 & -8 \end{bmatrix}$. Matriks A sebagai matriks koefisien – koefisien fungsi kendala, karena Contoh 2.3 hanya memiliki satu kendala maka matriks A menjadi matriks satu baris yaitu $A = \begin{bmatrix} 1 & 2 \end{bmatrix}$, sehingga dapat ditentukan $B = [30]$.

Setelah teridentifikasi, bentuk permasalahan dapat disusun ulang, yaitu :

$$f(x_1, x_2) = 15\,x_1 + 30\,x_2 + 4\,x_1x_2 - 2\,x_1{}^2 - 4x_2{}^2$$

$$= \begin{bmatrix} 15 & 30 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \frac{1}{2} \begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} -4 & 4 \\ 4 & -8 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + 0$$

atau,

$$f(X) = C^T X + \frac{1}{2} X^T Q X + d$$

dengan kendala :

$$x_1 + 2x_2 \leq 30$$

atau dapat ditulis,

$$\begin{bmatrix} 1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \leq [30]$$

atau,

$$AX \leq B.$$

Berdasarkan identifikasi yang telah dilakukan, maka bentuk pada Contoh 2.3 dapat diselesaikan dengan menggunakan pendekatan *quadratic programming*.

2. Penyelesaian *Quadratic Programming*

Permasalahan pada quadratic programming dapat diselesaiakan dengan menggunakan persyaratan Kuhn-Tucker seperti yang ditertera pada Teorema 2.3 dan Teorema 2.4. Selain itu, dalam *quadratic programming* juga terdapat kondisi *complementary slackness* khusus.

Secara umum, kondisi *complementary slackness* pada *quadratic programming* dapat dinyatakan dalam Sifat 2.1 berikut :

Sifat 2.1. *Complementary slackness* pada *quadratic programming* (Winston, 2003 : 687)

1) e_j dan s_j pada kondisi Kuhn-Tucker dan x_j tidak dapat kedua-duanya bernilai positif.
2) Variabel *surplus* (*excess*) ataupun *slack* untuk kendala ke-i dan λ_i tidak dapat kedua-duanya bernilai positif.

Bukti Sifat 2.1 :

1) Diperhatikan Syarat 1) dan 3) pada Teorema 2.3, yaitu :

   Syarat 1) yaitu : $\frac{\partial f}{\partial x_j} - \sum_{i=1}^{m} \lambda_i \frac{\partial g_i}{\partial x_j} + s_j = 0$, sehingga

   $\frac{\partial f}{\partial x_j} - \sum_{i=1}^{m} \lambda_i \frac{\partial g_i}{\partial x_j} = -s_j$ disubstitusikan ke Syarat 3)

   $(\frac{\partial f}{\partial x_j} - \sum_{i=1}^{m} \lambda_i \frac{\partial g_i}{\partial x_j})x_j = 0$

   $s_j x_j = 0.$

   Jika $s_j = 0$ maka $x_j \neq 0$, yaitu $x_j > 0$.

   Jika $x_j = 0$ maka $s_j \neq 0$, yaitu $s_j > 0$ atau $s_j < 0$. Berdasarkan Syarat 4) maka $s_j > 0$.

Hal ini berlaku juga untuk Teorema 2.4, sehingga terbukti bahwa e_j dan s_j pada kondisi Kuhn-Tucker dan x_j tidak dapat kedua-duanya bernilai positif.

2) Diperhatikan Syarat 2) yaitu $\lambda_i[\, b_i - g_i(\mathbf{x})] = 0$

   Pada fungsi kendala $g_i(\mathbf{x}) \le b_i$ maka bentuk kanonik kendala tersebut yaitu $g_i(\mathbf{x}) + s_i' = b_i$, sehinggan Syarat 2 menjadi :

   $\lambda_i s_i' = 0$

   Jika $\lambda_i = 0$ maka $s_i' \neq 0$, yaitu $s_i' > 0$.

   Jika $s_i' = 0$ maka $\lambda_i \neq 0$, yaitu $\lambda_i > 0$ atau $\lambda_i < 0$. Berdasarkan Syarat 5) maka $\lambda_i > 0$.

   Pada fungsi kendala $g_i(\mathbf{x}) \ge b_i$ dapat diubah menjadi $g_i(\mathbf{x}) - e_i' = b_i$.

   Melalui cara yang sama maka didapat pula $\lambda_i e_i' = 0$, sehingga terbukti bahwa variabel *surplus* (*excess*) ataupun *slack* untuk kendala ke-i dan λ_i tidak dapat kedua-duanya bernilai positif

Persamaan – persamaan yang didapat dari langkah a merupakan langkah pelinearan masalah pemrograman nonlinear dengan menggunakan syarat Kuhn Tucker, selanjutnya masalah tersebut dapat diselesaikan dengan substitusi atau cara – cara yang lain.

Contoh 2.4 berikut merupakan contoh masalah model nonlinear dengan *quadratic programming* :

Contoh 2.4:

Diketahui model nonlinear kuadratik yang meminimumkan

$$z = -x_1 - x_2 + \left(\frac{1}{2}\right) {x_1}^2 + {x_2}^2 - x_1 x_2 \tag{2.26}$$

dengan kendala

$x_1 + x_2 \leq 3$ (2.27a)

$-2x_1 - 3x_2 \leq -6$ (2.27b)

$x_1, x_2 \geq 0$

Penyelesaian :

Diperhatikan bahwa (2.26) dan (2.27) dapat diselesaikan dengan *quadratic programming* dengan langkah sebagai berikut :

a. Membentuk syarat Kuhn-Tucker dari model nonlinear.

Berdasarkan Teorema 2.4, maka pada Contoh 2.4 dapat ditentukan syarat Kuhn Tuckernya yaitu :

1) $-1 + x_1 - x_2 + \lambda_1 - 2\lambda_2 - e_1 = 0$ (2.28a)

$-1 + 2x_2 - x_1 + \lambda_1 - 3\lambda_2 - e_2 = 0$ (2.28b)

2) $\lambda_1[3 - (x_1 + x_2)] = 0$ (2.29a)

$\lambda_2[-6 - (-2x_1 - 3x_2)] = 0$ (2.29b)

3) $(-1 + x_1 - x_2 + \lambda_1 - 2\lambda_2)x_1 = 0$ (2.30a)

$(-1 + 2x_2 - x_1 + \lambda_1 - 3\lambda_2)x_2 = 0$ (2.30b)

4) $\lambda_1,\ \lambda_2 \geq 0$ (2.31)

5) $e_1, e_2 \geq 0$ (2.32)

Sebagai akibat dari (2.29) maka :

$x_1 + x_2 - 3 \leq 0$ (2.33a)

$(-2x_1 - 3x_2) - (-6) \leq 0$ (2.33b)

Bentuk (2.33) dapat dijadikan bentuk kanonik sehingga menjadi :

$x_1 + x_2 + s_1' = 3$ (2.34a)

$$2x_1 + 3x_2 - e_2{}' = 6 \quad (2.34b)$$

b. Mengidentifikasi *complementary slackness* yang ada.

Berdasarkan (2.29) dan (2.34), (2.28) dan (2.30), dan Sifat 2.1, maka *complementary slackness* untuk Contoh 2.4 adalah :

$$\lambda_2 e_2{}' = 0 \qquad \lambda_1 s_1{}' = 0 \qquad e_1 x_1 = 0 \qquad e_2 x_2 = 0$$

Model nonlinear tersebut selanjutnya diselesaikan dengan substitusi ataupun cara yang lain.

J. *Separable Programming*

1. Bentuk Umum Model Nonlinear untuk *Separable Programming*

Bentuk penyelesaian permasalahan nonlinear selanjutnya yaitu *separable programming*. Menurut definisi Desi Mariani (2003 : 3), *separable programming* adalah pemrograman tak linear (nonlinear) yang fungsi objektif (fungsi tujuan) dan fungsi kendalanya dapat diekspresikan sebagai penjumlahan fungsi dan setiap fungsinya hanya terdiri atas satu variabel.

S.S. Rao (1978 : 640) merumuskan bentuk umum model nonlinear yang diselesaikan dengan *separable programming* yaitu sebagai berikut :

Memaksimumkan / meminimumkan $f(X) = \sum_{j=1}^{n} f_j(x_j)$, (2.37a)

dengan kendala , $\sum_{j=1}^{n} g_{ji}(x_j)(\geq, \leq) b_i$ untuk setiap $i = 1,2, \dots, m$ (2.37b)

Fungsi tujuan yang dibentuk harus dipisahkan berdasarkan variabel. Contoh 2.5 berikut akan menjelaskan cara penyusunan fungsi *separable* untuk fungsi – fungsi khusus.

Contoh 2.5 (Rao, 1978 : 640 - 641)

1) Suatu fungsi nonlinear $f = x_1 x_2$ akan diubah menjadi fungsi *separable.* Langkah pertama yaitu membuat variabel baru berupa y_1 dan y_2, dengan :

$$y_1 = \frac{x_1 + x_2}{2}, \tag{2.39}$$

$$y_2 = \frac{x_1 - x_2}{2}. \tag{2.40}$$

Berdasarkan Persamaan (2.39) dan (2.40), maka

$$x_1 x_2 = \frac{1}{4}(x_1 + x_2)^2 - \frac{1}{4}(x_1 - x_2)^2 = y_1{}^2 - y_2{}^2. \tag{2.41}$$

Berdasarkan Persamaan (2.41), maka $f = x_1 x_2$ termasuk fungsi yang dapat dipisahkan dengan transformasi bentuk baru fungsi menjadi

$$f = y_1{}^2 - y_2{}^2.$$

2) Suatu masalah program nonlinear yaitu :

Meminimumkan $f = 5e^{(4x_1 + x_2)} + 10x_2{}^2$ (2.42)

dengan kendala $10x_1 x_2 + 15x_2{}^2 = 100$ (2.43)

$x_1, x_2 > 0$

akan diubah ke dalam bentuk *separable programming.*

Pada fungsi *f* tersebut x_1 dan x_2 tidak dapat dipisahkan secara langsung, sehingga perlu dibuat variabel baru untuk memisahkannya, yaitu

$y_1 = e^{(4x_1+x_2)}$ (2.44)

Sehingga $\ln y_1 = 4x_1 + x_2$ (2.45)

Fungsi kendala juga harus diubah menjadi bentuk *separable.* Pada Persamaan (2.43), $10x_1x_2$ diubah dengan membuat variabel baru yaitu :

$y_2 = x_1x_2,$ (2.46)

Sehingga $\ln y_2 = \ln x_1 + \ln x_2.$ (2.47)

Persamaan (2.44) dan (2.46) disubstitusikan ke persamaan (2.42) dan (2.43) sehingga diperoleh bentuk *separable* untuk Persamaan (2.42) dan (2.43) adalah :

Meminimumkan $f = 5y_1 + 10{x_2}^2$ (2.48)

dengan kendala $10y_2 + 15{x_2}^2 = 100$ (2.49a)

dan terdapat tambahan kendala baru berdasarkan (2.45) dan (2.47), yaitu :

$$\ln y_1 - 4x_1 - x_2 = 0 \quad (2.49b)$$

$$\ln y_2 - \ln x_1 - \ln x_2 = 0 \quad (2.49c)$$

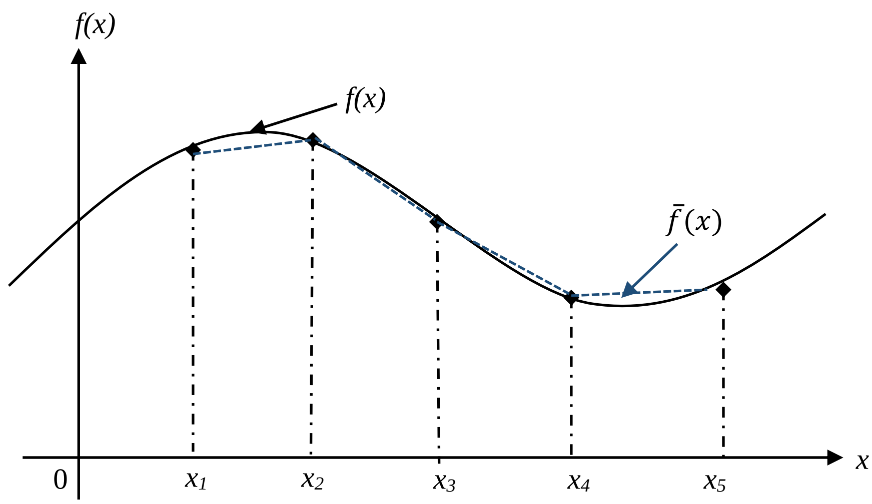
$$x_1, x_2 > 0$$

Berdasarkan Contoh 2.5 tersebut, disimpulkan bahwa banyak bentuk nonlinear dapat dijadikan fungsi *separable* dengan mentransformasikannya menjadi variabel baru yang dapat dipisahkan.

Pada penyelesaian permasalahan *separable programming* perlu diperhatikan bahwa fungsi tujuan masalah berpola meminimumkan yang dikerjakan merupakan suatu bentuk penjumlahan dari $f_j(x_j)$ yang berupa fungsi – fungsi cembung. Pada masalah berpola memaksimumkan, maka fungsi tujuan berupa jumlahan dari $f_j(x_j)$ yang berupa fungsi – fungsi cekung, sedangkan fungsi kendala selalu berupa fungsi cembung (Budi Marpaung, 2012).

2. Penyelesaian *Separable Programming*

Penyelesaian *separable programming* seringkali menggunakan hampiran fungsi linear sepenggal. Gambar 2.76berikut merupakan ilustrasi hampiran fungsi linear sepenggal untuk suatu fungsi *f(x)* dengan beberapa *grid point.*



Gambar 2. 6 Grafik hampiran fungsi linear sepenggal pada fungsi nonlinear

Pada Gambar 2.6, nilai $f(x)$ merupakan nilai sesungguhnya dari fungsi nonlinear, sedangkan $\bar{f}(x)$ adalah nilai hampiran fungsi linear sepenggal yang mana dapat dicari dengan rumus pendekatan berikut (Rao, 1978 : 642) :

$$\bar{f}(x) = f(x_1) + \left[\frac{f(x_2)-f(x_1)}{x_2-x_1}\right](x - x_1);\ x_1 \leq x \leq x_2 \qquad (2.50a)$$

$$\bar{f}(x) = f(x_2) + \left[\frac{f(x_3)-f(x_2)}{x_3-x_2}\right](x - x_2);\ x_2 \leq x \leq x_3 \qquad (2.50b)$$

$$\vdots$$

$$\bar{f}(x) = f(x_k) + \left[\frac{f(x_{k+1}) - f(x_k)}{x_{k+1} - x_k}\right](x - x_k);\ x_k \leq x \leq x_{k+1} \tag{2.50c}$$

Jika pembagian $\frac{x - x_1}{x_2 - x_1}$ dinyatakan sebagai λ, maka persamaan (2.50a) dapat ditulis menjadi

$$\bar{f}(x) = f(x_1) + \lambda\big(f(x_2) - f(x_1)\big) = \lambda f(x_2) + (1 - \lambda) f(x_1) \tag{2.51}$$

Selanjutnya, dengan memberi notasi baru untuk $1 - \lambda = \lambda_1$ dan $\lambda = \lambda_2$ maka persamaan (2.51) dapat diubah menjadi

$$\bar{f}(x) = \lambda_1 f(x_1) + \lambda_2 f(x_2);\ x_1 \leq x \leq x_2 \tag{2.52}$$

$$\lambda_1 + \lambda_2 = 1, \text{ dan } \lambda_1, \lambda_2 \geq 0 \tag{2.53}$$

Karena $\lambda = \frac{x - x_1}{x_2 - x_1}$ maka

$$x - x_1 = \lambda(x_2 - x_1)$$

$$x = x_1 + \lambda(x_2 - x_1)$$

$$x = (1 - \lambda)x_1 + \lambda x_2 = \lambda_1 x_1 + \lambda_2 x_2 \tag{2.54}$$

Persamaan (2.53) dan (2.54) juga berlaku untuk interval x_j yang lain, sehingga diperoleh bentuk umum yaitu : (Rao, 1978 : 644)

$$x = \sum_{k=1}^{p_i} \lambda_k x_k \text{ , dengan } p_i \text{ adalah jumlah titik interval} \tag{2.55}$$

$$\text{dengan } \sum_{k=1}^{p_i} \lambda_k = 1 \text{ dan } \lambda_k \geq 0\ ; k = 1,2, \dots, p_i \tag{2.56}$$

Persamaan (2.55) merupakan interpretasi baru untuk variabel keputusan yang akan diselesaikan dengan menggunakan hampiran fungsi linear sepenggal, dengan (2.56) sebagai tambahan fungsi kendala.

Adapun langkah – langkah penyelesaian *separable programming* dengan hampiran fungsi linear sepenggal yaitu sebagai berikut :

a. Menyusun fungsi kendala yang *separable.*

Bentuk kendala yang baru dapat disusun berdasarkan (2.37b) untuk sejumlah *m* kendala. Sehingga bentuk kendala dapat disusun menjadi :

$$g_{11}(x_1) + g_{21}(x_2) + \cdots + g_{n1}(x_n)\ (\geq, \leq) b_1$$

$$g_{12}(x_1) + g_{22}(x_2) + \cdots + g_{n2}(x_n)\ (\geq, \leq) b_2$$

$$\vdots$$

$$g_{1m}(x_1) + g_{2m}(x_2) + \cdots + g_{nm}(x_n)\ (\geq, \leq) b_m$$

b. Menentukan jumlah *grid point*

Grid point merupakan titik – titik bagi dari interval $a_j \leq x_j \leq b_j$. Batas a_j dan b_j menjadi batas bawah dan batas atas untuk setiap variabel x_j. Setiap variabel x_j dibagi lagi sejumlah p_i interval. Jika x_{kj} merupakan nilai x_j pada titik ke – *k*, maka dapat diperoleh bentuk $a_j = x_{1j} < x_{2j} < \cdots < x_{kj} < \cdots < x_{p_i j} = b_j$ (Rao, 1978 : 642). Notasi $x_{1j}, x_{2j}, \ldots, x_{p_i j}$ merupakan partisi nilai – nilai *x* yang dibagi menjadi p_i *grid point*. Jumlah *grid point* tersebut ditentukan sesuai kebutuhan dengan batas atas dan bawah tetap dimasukkan sebagai *grid point*, namun demikian semakin banyak *grid point* yang dibentuk maka semakin banyak variabel yang mucul dan solusi optimal yang dihasilkan semakin akurat. Adapun interval dari setiap *grid point* tidak harus berjarak sama.

c. Membentuk nilai fungsi *grid point*

Setiap nilai *grid point* kemudian disubstitusikan ke fungsi tujuan yang sudah dipisahkan $(f_j(x_j))$. Nilai yang didapatkan kemudian menjadi koefisien baru untuk fungsi tujuan linear. Hal ini juga berlaku untuk fungsi kendala, dimana setiap nilai grid point juga disubstitusikan pada fungsi kendala *separable* yang telah dibentuk pada langkah a.

d. Membentuk fungsi tujuan baru yang linear

Bentuk dari fungsi tujuan yang linear dari persamaan (2.37) adalah :

meminimumkan / memaksimumkan $W = \sum_{j=1}^{n} \sum_{k=1}^{p_i} f_{kj}\lambda_{kj}$ (2.57)

dengan p_i merupakan jumlah *grid point.*

Bentuk fungsi kendala menjadi

$\sum_{j=1}^{n} \sum_{k=1}^{p_i} g_{kji}\lambda_{kj} (\leq, \geq)\, b_i, i = 1, 2, \dots, m$ (2.58a)

$\sum_{k=1}^{p_i} \lambda_{kj} = 1, j = 1,2, \dots, n$ (2.58b)

$\lambda_{kj} \geq 0$, $k = 1, \dots, p_i$, dan $j = 1,2, \dots, n$ (2.58c)

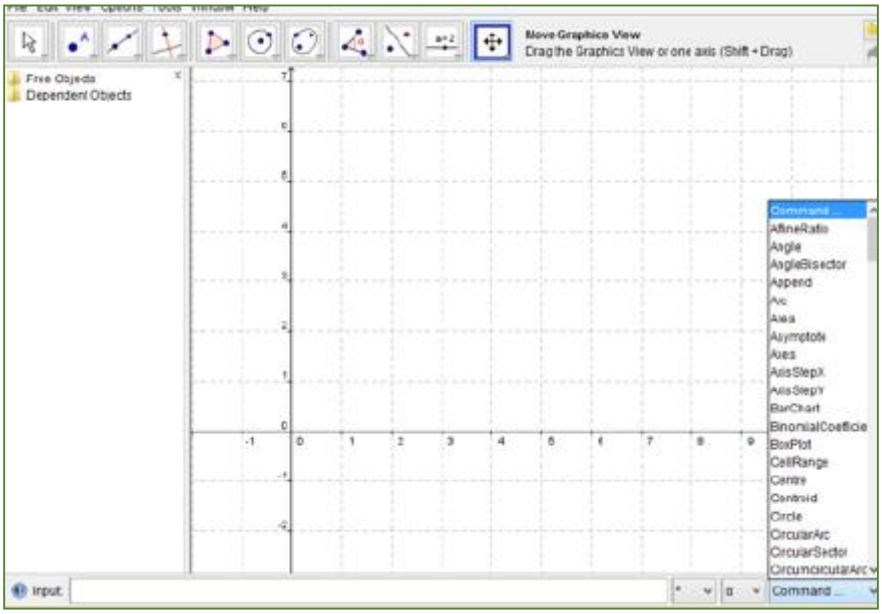
e. Menyelesaikan bentuk linear dengan metode simpleks.

Bentuk linear yang telah dibentuk selanjutnya dapat diselesaikan dengan menggunakan metode simpleks biasa seperti pada pemrograman linear.

K. *Software* Geogebra

Software Geogebra merupakan salah satu aplikasi komputer di bidang matematika yang menggabungkan geometri, aljabar, dan kalkulus. *Software* ini dilengkapi dengan fitur untuk menampilkan grafik dari sebuah fungsi. Gambar 2.7 dan 2.8 berikut merupakan tampilan saat membuka Geogebra.

Gambar 2. 7 Tampilan Pembuka Geogebra

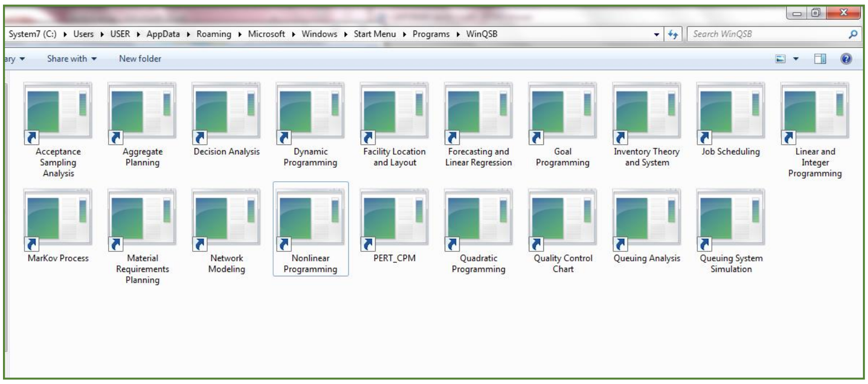


Gambar 2. 8 Tampilan Utama Geogebra

L. *Software* WinQSB 2.0

Software WinQSB 2.0 merupakan *software* aplikasi matematika untuk menyelesaikan masalah optimasi agar didapat solusi optimal. *Software* ini menyediakan beberapa menu pilihan untuk menyelesaikan kasus-kasus dengan kriteria khusus sesuai kebutuhan. Pada skripsi ini, pilihan yang akan digunakan adalah sub menu *software* WinQSB *Nonlinear Programming*, yaitu untuk mencari solusi optimal dari kasus optimasi dengan fungsi tujuan nonlinear. Pada *Nonlinear*

Programming untuk masalah model nonlinear berkendala yang dikerjakan WinQSB menggunakan metode *Penalty Function*, sehingga dapat dikatakan bahwa dalam penelitian ini akan membandingkan metode *quadratic programming*, *separable programming*, dan *penalty function.* Selain menggunakan sub menu *Nonlinear Programming*, juga digunakan sub menu lainnya yaitu *Linear and Integer Programming* untuk membantu perhitungan simpleks pada fungsi linear. Gambar 2.9 dan 2.10 berikut merupakan tampilan awal saat membuka WinQSB.



Gambar 2. 9 Tampilan Pilihan Sub Menu WinQSB

Gambar 2. 10 Tampilan Pembuka WinQSB 2.0 *Nonlinear Programming*