Jurnal Matematika Vol.6 No.1
Nopember 2006 [71 - 75]

PENENTUAN EOQ MASALAH PERSEDIAAN MULTI-ITEM DENGAN NONLINEAR GOAL PROGRAMMING

M.Yusuf Fajar, Erwin Harahap, Farid H Badruzzaman

Jurusan Matematika, UNISBA, Jalan Tamansari No 1, Bandung,40116, Indonesia

myusuffajar@yahoo.com, erwin2h@gmail.com, faridhbadruzzaman@yahoo.co.id

Abstrak. Makalah ini membahas bagaimana model Nonlinear *Goal Programming* dapat menentukan EOQ untuk masalah persediaan multi-item. Dalam proses mencari solusi, analisis sensitifitas dan prioritas dengan pembobotan diberikan. Diperoleh himpunan solusi, dari solusi-solusi tersebut diidentifikasikan solusi ideal. Dihitung jarak D_1 yaitu selisih dari solusi-solusi dengan solusi ideal. Solusi yang berhubungan dengan jarak D_1 minimum memberikan solusi kompromi yang terbaik.

Kata kunci : EOQ; *non linear programming*; *multi-item*

1. Pendahuluan

Dalam masalah pengambilan keputusan, terdapat beberapa tujuan yang saling konflik. Fungsi biaya yang akan diminimumkan merupakan fungsi non linier, sehingga model yang digunakan adalah model non linier *goal programming*. Untuk setiap fungsi tujuan, ditentukan target level yang ingin dicapai, prioritas dan pembobotannya. Solusi yang diperoleh merupakan solusi kompromi terbaik dari beberapa alternatif solusi yang diperoleh.

Dalam menentukan *Economic Order Quality* (EOQ), untuk masalah persediaan *multi-item*, fungsi biaya merupakan fungsi non-linier, sehingga meminimumkan fungsi biaya secara keseluruhan dilakukan dengan meminimumkan k fungsi biaya.

2. Formulasi Model

Model umum dari masalah persediaan multi-item adalah mencari $\overline{q} = (q_1, q_2, ..., q_n)$ sehingga

$$\text{meminimumkan } \sum_{i=1}^{n} (C_i q_i + \frac{C_{si}}{q_i}) \tag{1}$$

dengan kendala : $\sum_{i=1}^{n} a_{ij} q_i \leq=\geq b_j$, $1 \leq j \leq m$ dan $0 < q_i \leq Q_i$, $1 \leq i \leq n$

dengan C_1,C_2, … , C_n adalah ordering cost dan $C_{s1}, C_{s2}, ..., C_{sn}$ adalah set-up cost, dan $Q_1, Q_2, ..., Q_n$ masing-masing adalah tingkat persediaan dari $q_1, q_2, ..., q_n$.

Dari (1) biaya yang berhubungan dengan $q_1, q_2, ..., q_n$ merupakan tujuan yang akan diminimumkan dengan beberapa batasan-batasan. Karena biaya yang berhubungan dengan item-item saling bebas dan berinterrelasi dengan beberapa kendala sehingga meminimumkan seluruh fungsi tujuan terhadap kendala dapat diberlakukan sebagai *n* fungsi tujuan terhadap kendala yang diberikan. Jadi tujuan (1) dapat dianggap sebagai *n* fungsi tujuan. Dalam *goal programming* setiap $C_i q_i + \frac{C_{si}}{q_i}$ ditinjau sebagai *goal* dengan variable deviational d_{ik}^{-} dan d_{ik}^{+}, serta target level G_i , $1 \leq i \leq n$.

Model masalah persediaan multi-item adalah mencari $\overline{q} = (q_1, q_2, ..., q_n)$ sehingga

Meminimumkan $P_1 \left[\sum_i (w_{i1}^- d_{i1}^- + w_{i1}^+ d_{i1}^+) + \sum_j (w_{n+j,1}^- d_{n+j,1}^- + w_{n+j,1}^+ d_{n+j,1}^+) \right]$

Meminimumkan $P_2 \left[\sum_i (w_{i2}^- d_{i2}^- + w_{i2}^+ d_{i2}^+) + \sum_j (w_{n+j,2}^- d_{n+j,2}^- + w_{n+j,2}^+ d_{n+j,2}^+) \right]$

…………………………………………………………………………

Meminimumkan $P_k \left[\sum_i (w_{ik}^- d_{ik}^- + w_{ik}^+ d_{ik}^+) + \sum_j (w_{n+j,k}^- d_{n+j,k}^- + w_{n+j,k}^+ d_{n+j,k}^+) \right]$

…………………………………………………………………………

Meminimumkan $P_K \left[\sum_i (w_{iK}^- d_{iK}^- + w_{iK}^+ d_{iK}^+) + \sum_j (w_{n+j,K}^- d_{n+j,K}^- + w_{n+j,K}^+ d_{n+j,K}^+) \right]$

Dengan kendala : $C_i q_i + \dfrac{C_{si}}{q_i} + d_{ik}^- - d_{ik}^+ = G_i, \quad 1 \le i \le n$ (2)

$$\sum_{i=1}^{n} a_{ij} q_i + d_{n+j,k}^- - d_{n+j,k}^+ = b_j, \quad 1 \le j \le m$$

$$0 < q_i \le Q_i, \quad d_{ik}^- . d_{ik}^+ = d_{n+j,k}^- . d_{n+j,k}^+ = 0$$

$$d_{ik}^-, d_{ik}^+, d_{n+j,k}^-, d_{n+j,k}^+ \ge 0, \ 1 \le i \le n, \ 1 \le j \le m, \ 1 \le k \le K$$

Dengan $P_k (1 \le k \le K : K \le n+m)$ adalah prioritas ke-*k* dari fungsi tujuan. *Goal* dengan prioritas yang lebih tinggi diutamakan, baru kemudian disusul dengan *goal* yang prioritasnya lebih rendah. Bobot variabel deviational $d_{ik}^-, d_{ik}^+, d_{n+j,k}^-, d_{n+j,k}^+$ adalah $w_{ik}^-, w_{ik}^+, w_{n+j,k}^-, w_{n+j,k}^+ \ge 0$.

3. Prosedur solusi

Misalkan $\left\{ q_1^{(r)}, q_2^{(r)}, ..., q_n^{(r)} \right\}$, $1 \le r \le K!$ adalah K! solusi yang diperoleh dari permutasi K prioritas.

Misalkan untuk *i*=*n* , $\min\limits_{1 \le r \le K!} \left(C_n q_n^{(r)} + \dfrac{C_{sn}}{q_n^{(r)}} \right) = \min\limits_{1 \le r \le K!} \left(C_n q_n^* + \dfrac{C_{sn}}{q_n^*} \right)$ dengan q_n^* adalah suatu nilai dari $q_n^{(r)}$, $1 \le r \le K!$, maka solusi ideal q^* didefinisikan oleh $q^* = \left\{ q_1^*, q_2^*, ..., q_n^* \right\}$. Tetapi dalam prakteknya solusi ideal tidak pernah dicapai. Solusi yang dekat dengan solusi ideal, diterima sebagai solusi kompromi terbaik.

Untuk memperoleh solusi kompromi terbaik, diselesaikan masalah *goal programming* berikut ini :

$$\min_{1 \le r \le K!} \sum_{i=1}^{n} \left(d_{ir}^+ + d_{ir}^- \right)$$

Dengan kendala : $q_i^* - q_i^{(r)} + d_{ir}^- + d_{ir}^+ = 0, \quad 1 \le r \le n$

$$d_{ir}^{+} \geq 0, d_{ir}^{-} \geq 0, \quad d_{ir}^{+}.d_{ir}^{-} = 0, \quad 1 \leq r \leq K!, \quad 1 \leq i \leq n$$

Dengan d_{ir}^{+}, d_{ir}^{-} adalah variabel deviational.

Selanjutnya didefinisikan jarak D_1 sebagai berikut :

$$(D_1)^r = \sum_{i=1}^{n} \left| q_i^* - q_i^{(r)} \right|$$

Merupakan jarak dari solusi ideal $q^* = \left\{ q_1^*, q_2^*, ..., q_n^* \right\}$ ke solusi ke-*r* $\left\{ q_1^{(r)}, q_2^{(r)}, ..., q_n^{(r)} \right\}$, $1 \leq r \leq K!$.

Oleh karena itu,

$$\left(D_1\right)_{opt} = \min_{1 \leq r \leq K!} \left(D_1^{(r)} \right) = \min_{1 \leq r \leq K!} \sum_{i=1}^{n} \left| q_i^* - q_i^{(r)} \right|$$

$$= \min_{1 \leq r \leq K!} \sum_{i=1}^{n} \left(d_{ir}^{+} + d_{ir}^{-} \right) \text{, katakanlah}$$

$$= \left(D_1 \right)^p, \quad 1 \leq p \leq K!$$

Jadi $\left\{ q_1^{(p)}, q_2^{(p)}, ..., q_n^{(p)} \right\}$ adalah solusi kompromi terbaik .

4. Studi Kasus

Ditinjau suatu kasus pada suatu perusahaan yang menyimpan berbagai jenis barang di gudang. Permintaan untuk setiap item diasumsikan seragam untuk tiap periode. Total anggaran adalah 0,9 juta dollar. Kapasitas gudang 368 unit. Dalam memilih item-item, manajemen memilih 12 item yang dikirim oleh 3 *supplier* dengan data sebagai berikut :

Tabel 1 : Data untuk semua item

Supplier	Item	C_i \$/unit	C_{si} \$	Q_i unit	Rata-rata investasi \$/unit	G_i \$
1	1	500	27600	80	550	38000
	2	600	15000	40	650	20000
	3	700	30000	20	750	14000
	4	400	26000	30	460	10000
2	5	1400	68000	60	1600	80000
	6	1700	69000	50	1950	86000
	7	1200	45000	40	1400	40888
3	8	4000	62000	40	4500	160000
	9	6500	103000	30	7500	200000
	10	12700	138000	20	14000	260000
	11	900	40000	30	1200	25000
	12	1600	61000	20	1800	20000

Dari data diatas, diperoleh persamaan sebagai berikut :

$$5q_1 + \frac{276}{q_1} + d_{1k}^- - d_{1k}^+ = 380 \text{, (item 1) ; } 6q_2 + \frac{150}{q_2} + d_{2k}^- - d_{2k}^+ = 200 \text{ , (item 2)}$$

$$7q_3 + \frac{300}{q_3} + d_{3k}^- - d_{3k}^+ = 140 \text{, (item 3) ; } 4q_4 + \frac{260}{q_4} + d_{4k}^- - d_{4k}^+ = 100 \text{ , (item 4)}$$

$$14q_5 + \frac{680}{q_5} + d_{5k}^- - d_{5k}^+ = 800 \text{, (item 5) ; } 17q_6 + \frac{690}{q_6} + d_{6k}^- - d_{6k}^+ = 860 \text{, (item 6)}$$

$$12q_7 + \frac{450}{q_7} + d_{7k}^- - d_{7k}^+ = 480 \text{, (item 7) ; } 40q_8 + \frac{620}{q_8} + d_{8k}^- - d_{8k}^+ = 1600 \text{, (item 8)}$$

$$65q_9 + \frac{1030}{q_9} + d_{9k}^- - d_{9k}^+ = 2000 \text{, (item 9) ;} 127q_{10} + \frac{1380}{q_{10}} + d_{10k}^- - d_{10k}^+ = 2600 \text{, (item 10)}$$

$$9q_{11} + \frac{400}{q_{11}} + d_{11k}^- - d_{11k}^+ = 250 \text{, (item 11) ; } 16q_{12} + \frac{610}{q_{12}} + d_{12k}^- - d_{12k}^+ = 200 \text{, (item 12)}$$

$$\sum_{i=1}^{12} q_i + d_{13k}^- - d_{13k}^+ = 368 \text{ (kapasitas gudang)}$$

$$55q_1 + 65q_2 + 75q_3 + 46q_4 + 160q_5 + 195q_6 + 140q_7 + 450q_8 + 750q_9 + 1400q_{10} + 120q_{11} + 180q_{12} + d_{14k}^- + d_{14k}^+ = 90000$$

Struktur pembobotan adalah :

$$S_K^1 \quad : \quad \left(3d_{1k}^+ + 2d_{4k}^+ + 2d_{6k}^+ + d_{8k}^+\right)$$

$$S_K^2 \quad : \quad \left(2d_{2k}^+ + d_{3k}^+ + 3d_{5k}^+ + 3d_{7k}^+ + d_{9k}^+ + d_{10k}^+ + 2d_{11k}^+ + d_{12k}^+\right)$$

$$S_K^3 \quad : \quad \left(d_{13k}^- + d_{14k}^-\right)$$

K adalah prioritas (K=1,2,3) . Terdapat 3! = 6 masalah yang akan diselesaikan. Semua masalah pemrograman nonlinear diselesaikan melalui program computer dengan hasil sebagai berikut :

Tabel 2 : Solusi

Run	Prioritas	Solusi											
		q_1	q_2	q_3	q_4	q_5	q_6	q_7	q_8	q_9	q_{10}	q_{11}	q_{12}
1	$S_1^1 S_2^2 S_3^3$	76,04	33,29	**12,2**	21,79	55,97	44,84	34,26	34,29	**22,01**	**13,07**	**22,17**	**9,71**
2	$S_1^1 S_2^3 S_3^2$	76,12	33,4	12,22	21,88	56,08	44,94	34,36	34,34	24,73	14,22	22,96	9,74
3	$S_1^2 S_2^1 S_3^3$	77,68	36,44	15,27	25,73	56,24	47,42	36,86	36,72	25,95	15,68	26,17	11,62
4	$S_1^2 S_2^3 S_3^1$	77,69	36,24	15,07	25,53	56,04	47,52	36,66	36,52	25,75	15,47	25,97	11,42
5	$S_1^3 S_2^1 S_3^2$	**75,19**	**33,08**	12,88	**20,66**	**54,00**	**43,81**	**34,11**	**34,13**	23,66	13,72	22,95	9,73
6	$S_1^3 S_2^2 S_3^1$	75,74	33,98	13,31	21,29	54,06	44,48	34,90	34,92	24,36	14,02	23,71	10,08
Solusi ideal		**75,19**	**33,08**	**12,20**	**20,66**	**54,00**	**43,81**	**34,11**	**34,13**	**22,01**	**13,07**	**22,17**	**9,71**

Selanjutnya dihitung jarak D_1 dari semua solusi yang mungkin, diperoleh hasil sebagai berikut :

Tabel 3 : Jarak D_1 dari semua solusi yang mungkin

Run	Jarak D_1
1	5,5
2	10,2
3	37,01
4	35,11
5	3,88
6	10,68

Dari Tabel 3 terlihat bahwa jarak D_1 minimum adalah 3,88. Oleh karena itu, solusi kompromi terbaik adalah :

$$q_1 = 75{,}19,\quad q_2 = 33{,}08,\quad q_3 = 12{,}88,\quad q_4 = 20{,}66,\quad q_5 = 54,\quad q_6 = 43{,}38$$
$$q_7 = 34{,}11,\quad q_8 = 34{,}13,\quad q_9 = 23{,}66,\quad q_{10} = 13{,}72,\quad q_{11} = 22{,}95,\quad q_{12} = 9{,}83$$

5. Kesimpulan

Metode ini selalu memperoleh solusi kompromi terbaik. Apabila solusi kompromi terbaik tidak sesuai dengan keinginan pengambil keputusan, maka dapat dipilih solusi kompromi terbaik berikutnya dan demikian seterusnya sampai diperoleh solusi yang memuaskan pihak manajemen. Proses dapat dilanjutkan sampai K! kali, karena K prioritas membentuk K! masalah dan pengambil keputusan mempunyai K! pilihan untuk memilih salah satu diantaranya.

Pustaka

[1] Basu M, Banerjee K, (1997), A Solution Procedure for Solving Multi-item Inventory Problem, International Journal of Mangement and System.

[2] Charnes A and Chooper W, (1961), *Management Models and Industrial Application of Linear Programming*, John Willey & Sons.Inc,New York.

[3] Ignizio JP, (1976), *Goal Programming and Extensions*, Lexington Books, Massachusettes.