PENERAPAN DINAMIK PROGRAMMING (DP)
dalam INDUSTRI & BISNIS

Oleh :
Ngarap Im Manik
Jurs. Matematika Binus-University

PENGERTIAN DP

Dinamik Programming/ Multi stage programming (DP) berciri memecah persoalan menjadi bagian yang lebih kecil (*sub problem* atau *stage*) di mana keputusan dibuat secara berurutan, shg keputusan pada satu tahap mempengaruhi keputusan berikutnya.

DP diperkenalkan *Richard Bellman* (1950)yg menyatakan:
***suatu kebijakan optimal mempunyai sifat bahwa apa pun keadaan dan keputusan awal, keputusan berikutnya harus membentuk suatu kebijakan optimal dgn memperhatikan keadaan dari hasil keputusan pertama*.**

Prinsip ini mempunyai arti bahwa :
Kita diperkenankan untuk mengambil keputusan yang layak bagi tahap persoalan yg masih tersisa, dan dalam rangkaian keputusan yg telah di ambil, hasil dari masing-masing tergantung pada hasil keputusan sebelumnya dalam rangkaian.Prosedur pemecahan persoalan dlm DP dilakukan secara rekursif.

CIRI POKOK Masalah DP

DP ialah suatu pendekatan matematika tentang optimisasi proses banyak tahap. Ciri dasar masalah DP adalah :

1. Keputusan tentang suatu masalah ditandai dgn optimisasi pada tahap berikutnya, ini berarti jika suatu masalah akan diselesaikan dgn DP ia dipisahkan menjadi n subproblem.
2. DP berkaitan dgn masalah di mana pilihan atau keputusan dibuat pada masing-masing tahap. Seluruh kemungkinan pilihan dicerminkan, di atur oleh *sistem status* atau *state* pada setiap tahap.
3. Setiap keputusan pada setiap tahap adalah *returnfunction* yg mengevaluasi pilihan yg dibuat.
4. Pada setiap tahap proses keputusan dihubungkan dgn tahap yg berdekatan melalui *fungsi transisi.* Fungsi ini berupa kuantitas yg diskrit maupun kontinu tergantung pada sifat masalah.
5. Suatu hubungan rekursif digunakan untuk menghubungkan kebijaksanaan optimum pada tahap *n* dengan *n-1.*

Prosedur rekursif dalam hal ini ada 2;

- Foreward recursive
- Backward recursive

Foreward recursive equation

$f_0 (X_0) = 0$
$f_j^*(X_j) = \text{opt}\,\{R_j(k_j) @ f_{j-1}^* (X_j @ k_j)\}$

Backward recursive equation

$f_n(Y_n) = 0$

$f_j^*(Y_j) = \text{opt}\{R_j(k_j) @ f_{j+1}^*(Y_j @ k_j)\}$

catatan: X atau Y =state;
j = tahap ke X @ k atau Y @ k = fungsi transisi
Simbol @ menyatakan hubungan matematik antara Xj atau Yj dengan kj, mis *tambah, kurang, kali atau akar, dll.*

6.Sekali kebijaksanaan optimum tahap *n* telah ditemukan, *n* komponen keputusan dapat ditemukan kembali dengan melacak balik melalui fungsi transisi tahap *n*.

Dalam perumusan DP terdapat tiga elemen pokok, yaitu :

- Stage,
- Alternatif (decision variabel) pada setiap stage & f.tujuan
- State sistem pada setiap stage.

Stage; adalah bagian persoalan yang mengandung decision variable
Alternatif; var keputusan yang ada pd setiap stage dan fungsi tujuan.
State; menunjukkan kaitan satu stage dengan stage yg lainnya, sedemikian rupa sehingga setiap stage dapat di optimisasikan secara terpisah, shg hasil optimisasi layak untuk seluruh persoalan.

Disisi lain state memudahkan membuat keputusan optimum bagi stage yg masih tersisa dengan tidak usah mengecek pengaruh keputusan yg akan datang pada keputusan yg dibuat sebelumnya.

State sulit diberikan definisi, tetapi kuncinya dapat dijumpai dengan menanyakan dua pertanyaan berikut :

1.Hubungan apa yang mengikat satu stage dengan stage lainnya.
2.Informasi apa yg diperlukan utk membuat keputusan yg layak pada stage yg sedang berlangsung tanpa mengecek keputusan layak yg telah dibuat stage sebelumnya.

PENERAPAN – DP

Berbagai persoalan praktis dapat diselesaikan dengan DP secara lebih efisien dari pada metode LP, tetapi tidak berarti bahwa DP selalu lebih efisien dari LP.
Berikut akan disajikan beberapa contoh persoalan yang diselesaikan dgn metode DP

Masalah Alokasi

Dalam contoh ini keputusan optimum dalam setiap tahap diperoleh dengan menggunakan metoda optimisasi klasik biasa.Keuntungan pada 4 macam kegiatan merupakan fungsi jam kerja yg dialokasikan pada masing-masing kegiatan seperti tabel. Jika setiap hari tersedia 4 jam kerja, bagaimana alokasi waktu sehingga keuntungan per hari maksimum ?

Jam kerja	K e g i a t a n			
	1	2	3	4
0	0	0	0	0
1	1	2	3	2
2	3	5	7	5
3	6	8	10	8
4	9	11	12	10

Misal 4 keputusan merupakan 4 state, variabel keputusan X_j(j=1,2,3,4) adalah banyaknya jam kerja yg dialokasikan pd tahap ke-j. $P_j(X_j)$ adalah keuntungan dari alokasi X jam kerja j, sehingga masalah dapat diformulasikan sebagai berikut:

Maks $z=P_1(X_1)+P_2(X_2)+P_3(X_3)+P_4(X_4)$
Kendala: $X_1+X_2+X_3+X_4 = 4$ dan $X_1,X_2,X_3,X_4 \geq 0$

Karena semua nilai var.keputusan diskrit, maka digunakan *backward recursive equation approach.* Misal
Y_1=jlh jam kerja alokasi pd tahap 1,2,3,4 ; Y_2=jlh jam kerja alokasi pd tahap 2,3,4
Y_3=jlh jam kerja alokasi pd tahap 3,4 ; Y_4=jlh jam kerja alokasi pd tahap 4
$f_4^*(Y_4)$=optimum profit tahap 4 dgn Y_4 tertentu. $f_3^*(Y_3)$=optimum profit tahap 3,4 dgn Y_3 tertentu.
$f_2^*(Y_2)$=optimum profit tahap 2,3,4 dgn Y_2 tertentu. ; $f_1^*(Y_1)$=optimum profit tahap 1,2,3,4 dgn Y_1 tertentu.
Tahap 4: $f_4^*(Y_4)$=maks{P_4X_4}, utk $f_5(Y_5)=0$

Y_4 \ X_4	$P_4(X_4)$ 0	1	2	3	4	$f_4^*(Y_4)$	X_4^*
0	0	-	-	-	-	0	0
1	-	2	-	-	-	2	1
2	-	-	5	-	-	5	2
3	-	-	-	8	-	8	3
4	-	-	-	-	10	10	4

Tahap 3: $f_3^*(Y_3)$=maks {$P_3(X_3) + f_4^*(Y_4)$},

Y_3 \ X_3	$P_3(X_3)+f_4^*(Y_4)$ 0	1	2	3	4	$f_3^*(Y_3)$	X_3^*
0	0	-	-	-	-	0	0
1	2	3	-	-	-	3	1
2	5	5	7	-	-	7	2
3	8	8	9	10	-	10	3
4	10	11	12	12	12	12	2,3,4

Tahap 2: $f_2^*(Y_2)$=maks {$P_2(X_2) + f_3^*(Y_3)$},

Y_2 \ X_2	$P_2(X_2)+f_3^*(Y_3)$ 0	1	2	3	4	$f_2^*(Y_2)$	X_2^*
0	0	-	-	-	-	0	0
1	3	2	-	-	-	3	0
2	7	5	5	-	-	7	0
3	10	9	8	8	-	10	0
4	12	12	12	11	11	12	0,1,2

Tahap 1: $f_1^*(Y_1)$=maks {$P_1(X_1) + f_2^*(Y_2)$},

Y_1 \ X_1	$P_1(X_1)+f_2^*(Y_2)$ 0	1	2	3	4	$f_1^*(Y_1)$	X_1^*
0	0	-	-	-	-	0	0
1	3	1	-	-	-	3	0
2	7	4	3	-	-	7	0
3	10	8	6	6	-	10	0
4	12	11	10	9	9	12	0

Tabel akhir keuntungan maksimum adalah 12.

Masalah Muatan (Knapsack/Cargo)

Misal perusahaan angkutan mempertimbangkan untuk mengangkut 3 jenis barang.Berat masing2 barang & biaya angkut sbb:

Barang (i)	Berat/item(w_j)	Biaya/item(v_j)
1	2	65
2	3	80
3	1	30

Jika armada perusahaan tsb memiliki kapasitas maks. W=5 ton, barang apa yg harus diangkut & berapa banyak agar diperoleh penerimaan maks?

Solusi:

Formulasi DP dibentuk dengan terlebih dahulu mengidentifikasikan tiga unsur dasar yaitu:

1.Tahap (j). Masing-masing barang meruapakan tahap.

2.State (Yj) adalah jlh berat angkut yg disediakan tahap ke n, n-1, ….j dimana Yj+1 = Yj – Wjkj; jadi Y1=W dan Yj=0,1,..W untuk j=2,3,…n

3.Var keputusan (kj) adalah banyak barang pada tahap j. Nilai Kj dapat lebih kecil 0 atau sebessar (W/wj) dan Kj integer.

Jika fj(Yj) = penerimaan optimum pada tahap n,n-1,…….j, maka ***Backward Recursive Equation*** sbb:

fn * (Yn) = Maks {vn kn} karena f n+1* (Yn+1) = 0

fj * (Yj) = Maks { vj kj + f *j+1 (Yj+1)}

kj = 0,1,….(Yj/wj) ; Yj = 0,1,…….W

Untuk masalah di atas dapat dibuat tabel sebagai berikut:

Tahap3: f_3*(Y_3)=maks{30k_3}, k_3=0,1,…5 &Y_3 = 0,1,..5

k_3 / Y_3	**30 k3**						**f_3*(Y_3)**	**k_3***
	0	1	2	3	4	5		
0	0	-	-	-	-	-	0	0
1	0	30	-	-	-	-	30	1
2	0	30	60	-	-	-	60	2
3	0	30	60	90	-	-	90	3
4	0	30	60	90	120	-	120	4
5	0	30	60	90	120	150	150	5

Tahap2: f_2*(Y_2)=maks{80k_2 **+** f_3* (Y_3)},
k_2=0,1 &Y_2 = 0,1,..5

k_2 / Y_2	80k_2+f_3*(Y_3)		**f_2*(Y_2)**	k_2*
	0	1		
0	0	-	0	0
1	30	-	30	0
2	60	-	60	0
3	90	80	90	0
4	120	110	120	0
5	150	140	150	0

Tahap 1: f_1*(Y_1)=maks{65k_1 **+** f_2* (Y_2)},
k_2=0,1,2 & Y_2 = 5

Y_1 \ k_1	$65k_1+f_2^*(Y_2)$			$F_1^*(Y_1)$	K_1^*
	0	1	2		
5	150	155	160	160	2

Solusi optimumnya adalah :
mengangkut barang 1 sebanyak 2 (k1* = 2)
mengangkut barang 2 sebanyak 0 (k2* = 0)
mengangkut barang 3 sebanyak 1 (k3* = 1) ; dengan penerimaan total = 160

SUMBER BACAAN

Bronson, Richard, 1991, Theory and Problem of Operations Research, USA, Schaum Outline Series, McGraw-Hill.

Mitchel G.H, 1992, *Operational Research Techniques and Example*, London, English University Press.

Siagian P, 1997, *Penelitian Operasional*, Jakarta, UI Press.

Taha A.Hamdy, 2003, *Operational Research*, USA, Pearson Education Inc, PHI.

Wagner, Harvey M,1995, *Principles of Operation Research with Apllication to Managerial Decisions* New Delhi, Prentice Hall Inc.