

MANAJEMEN PENGECEKAN INVENTARIS PERUSAHAAN BERBASIS PROGRAM DINAMIS

Ibnu Hikam

*Laboratorium Ilmu dan Rekayasa Komputasi
Departemen Teknik Informatika, Institut Teknologi Bandung
Jl. Ganesha 10, Bandung
e-mail: if15038@students.if.itb.ac.id*

ABSTRAK

Prinsip dunia industri adalah bagaimana mengoptimalkan output yang dihasilkan suatu perusahaan dengan biaya yang minimum. Sejalan dengan prinsip ekonomi yaitu mendapatkan untung yang maksimum dengan pengorbanan yang seminimumnya. Maka pengelolaan barang atau inventaris perusahaan dapat menjadi salah satu solusi bagaimana meminimumkan biaya yang dikeluarkan oleh perusahaan. Pada makalah ini akan dibahas pengecekan inventaris perusahaan dengan menggunakan pendekatan program dinamis atau *dynamic programming*. Dimana pada program dinamis terdapat karakteristik rekursif, dimana proses rekursif ini mengoptimalkan atau meminimumkan biaya yang dikeluarkan setiap tahap atau *state*-nya dalam pengelolaan pengecekan inventaris perusahaan. Untuk mempermudah pendekatan rumus dan numerik digunakan suatu kasus pada manajemen inventaris pada perusahaan *Jeppesen Sandersen, Inc* pada makalah ini.

Kata kunci: program dinamis, *boundary*, *interior*, *fixed cost*, *variable cos*, *plate*

1. PENDAHULUAN

Dalam dunia bisnis, khususnya industri setiap petarung bisnis pasti mengorientasikan bisnisnya untuk mendapatkan keuntungan semaksimal mungkin dengan pengorbanan seminimal mungkin. Salah satu cara untuk mendapatkan keuntungan semaksimal mungkin adalah dengan melakukan mimalisisir penganggaran biaya pada proses operasi produksinya salah satunya operasi pengecekan inventarisnya. Oleh karena itu, tiap instansi bisnis dituntut untuk memiliki suatu strategi optimasi untuk mengelola dan memenej proses pengecekan gudanya, Sebagai algoritma yang memecahkan masalah dengan

menguraikan solusi menjadi sekumpulan langkah,

Pemrograman Dinamis (*Dynamic Programming*) dan karakteristik rekursifnya merupakan algoritma yang tepat untuk menyelesaikan permasalahan optimasi tersebut.

hingga kini persoalan ini banyak sekali ditemukan dalam permasalahan sehari-hari, misalnya memilih barang apa saja yang akan dimasukkan ke dalam koper untuk dibawa bepergian, sampai pada masalah yang akan kita bahas, pengecekan inventaris perusahaan.

Dalam persoalan ini, kita ambil contoh penerapan sistem ini pada perusahaan yang bernama *jeppesen sanderson, Inc*. Perusahaan yang bergerak dibidang penyedia produk-produk yang berkaitan dengan teknologi informasi. Dimana *Jeppesen* mengoptimalisasi manajemen pengelolaan inventarisnya sehingga mampu memangkas pengeluarannya sampai 800.000 USD tiap tahunnya mulai dari tahun 1998..

2. PENGECEKAN INVENTARIS DENGAN PROGRAM DINAMIS

2.1 Definisi Program Dinamis

Dynamic Programming (Program Dinamis) adalah metode pemecahan masalah dengan cara menguraikan solusi menjadi sekumpulan langkah (*step*) atau tahapan (*stage*) sedemikian sehingga solusi dari persoalan dapat dipandang dari serangkaian keputusan yang saling berkaitan.

Dynamic Programming biasanya digunakan untuk memecahkan masalah optimasi, yakni masalah yang memiliki banyak kemungkinan solusi, dan *Dynamic Programming* diminta untuk mencari solusi yang paling optimal.

2.2 Jenis-Jenis Program Dinamis

Dalam menyelesaikan persoalan dengan *Dynamic Programming*, kita dapat menggunakan dua pendekatan berbeda,

Misalkan x_1, x_2, \dots, x_n menyatakan peubah (*variable*) keputusan yang harus dibuat untuk masing-masing tahap 1, 2, ..., n. Maka :

a. *Dynamic Programming Forward*, (Program Dinamis Maju) program dinamis bergerak mulai dari tahap 1, ke tahap 2, 3 dan seterusnya sampai tahap n. Runtunan *variable* keputusan adalah x_1, x_2, \dots, x_n .

b. *Dynamic Programming Backward*, (Program Dinamis Mundur) program dinamis bergerak dari tahap ke n, terus mundur ke tahap n-1, tahap n-2 sampai tahap 1. Pada Permasalahan ini kita akan menggunakan Program Dinamis Maju.

2.3 Formulasi Program Dinamis

Kita modelkan persoalan pengecekan inventaris *Jeppesen* sebagai sistem pengecekan inventaris periodik, dengan durasi periode acak (R). Dimana tiap R mewakili selang waktu pada grafik. Sehingga bisa ditemukan banyak sekali kemungkinan.

Namun secara sederhana kita dapat mengasumsikan selang waktu acak(R) dengan nilai kt , dimana k adalah angka integer yang merepresentasikan selang waktu dari siklus pengecekan, dan $t = 1, 2, 3, \dots, b$ yang melambangkan indeks siklus dengan b sebagai batas atas.

Sebagai contoh kita masukkan nilai $k = 8$ minggu dan $b = 6$ kali pengecekan, sehingga kt -nya adalah 8, 16, 24, 32, 40, 48. Yang berarti pengecekan inventaris dilakukan setiap 8 minggu sekali yaitu pada minggu ke 8, minggu ke 16, minggu ke 24, minggu ke 32, minggu ke 40 dan minggu ke 48.

Variabel yang lain adalah jumlah permintaan barang setiap minggunya dilambangkan oleh D_i , dimana $i = 1, 2, 3, \dots, b$. Kita sederhanakan saja bahwa jumlah permintaan barang setiap minggunya sama yaitu D. sehingga kita dapatkan D_i di buat sama dilambangkan dengan D.

Pengecekan inventaris memiliki dua biaya, sebelum hasil pengecekan dapat dipublikasikan (entah untuk kalangan sendiri ataupun ke masa umum) sebuah *plate* harus dibuat terlebih dahulu, dengan satu atau lebih gambaran negatif tentang hasil pengecekan yang akan dipublikasikan. Membuat *plate* pun menghabiskan waktu proses untuk pengerjaannya, yang kita jadikan sebagai biaya tetap atau *fixed cost* (A) pada tiap pemesanan barang. Setelah *plate* selesai dibuat maka sisa biaya pengecekan inventaris kita sebut c, yang merupakan biaya variabel tiap unit barang yang dipesan. Sehingga dapat kita definisikan fungsi $c(y)$ sebagai berikut.

$$c(y) = \begin{cases} A + cy & \text{when } y \geq 0, \\ 0 & \text{when } y = 0. \end{cases}$$

Agar menjadi sederhana, kita asumsikan dapat diisi secara *instant*. Dalam prakteknya, *vendor* sangat responsif, dan melakukan pemesanan tiap minggu.

Di *Jeppesen*, pengelolaan barang inventaris diabaikan, sehingga kita biarkan nilainya nol. Sehingga kita asumsikan bahwa pemesan harus memesan ulang jika inventaris sedang kosong atau nol. Sehingga A merupakan nilai biaya penalti jika pemesanan dilakukan dengan waktu yang singkat atau mendadak. Sehingga kita tidak menggunakan *variabel cost* pada contoh ini.

Kita formulasikan masalah *Jeppesen* ini sebagai masalah program dinamis atau *dynamic programming* (DP), dilambangkan oleh n, sebagai usia waktu pengecekan, yang dihitung sejak permintaan dilayangkan. Dapat diformulasikan sebagai $n = kt + i, t = 0, 1, 2, \dots, b-1$ dan $i = 1, \dots, k$. Jika usia pengecekan sekarang n, maka *state system* adalah x, yang mana merupakan awal permulaan inventaris pada periode itu.

Tujuan utama kita adalah dapat meminimalisir biaya selama proses pengecekan barang. Kita rumuskan menjadi $V_n(x)$ menjadi biaya minimum yang diharapkan dari usia pengecekan inventaris n sampai akhir periode. $V_1(0)$ merepresentasikan biaya minimum yang diharapkan dari total waktu yang dialokasikan pada pengecekan barang atau inventaris.

Kita sebut periode *interior* jika hasil pengecekan tidak bisa di kaji pada akhir periode. Periode *interior* memiliki indeks waktu $n = kt + i$, $i = 1, 2, 3 \dots, k-1$. Kita sebut periode *boundary* jika pengecekan berakhir. Dengan kata lain periode *boundary* adalah periode terakhir dalam siklus pengecekan tiap indeks $k(t + 1)$, $t = 0, 1, 2, \dots, b-1$. Dalam kasus *Jeppesen*, periode *boundary* adalah 8, 16, 24, 32, 40, dan 48, dan periode yang lain adalah periode *interior*. Yaitu (1,2, ..., 7, 9, ..., 15, 17, ..., 23, dan selanjutnya).

Kita notasikan sisa waktu dari pengecekan, dengan usia kt , yaitu

$$R_{kt} = R - kt \quad | R > kt,$$

$$t = 1, 2, \dots, b-1.$$

Lalu peluang kondisi pengecekan yang berlangsung, dengan usia kt , menyisakan siklus waktu t' , yang dihasilkan dari

$$P(R_{kt} = kt') = P(R = k(t + t')) \quad | R > kt,$$

$$t = 1, 2, \dots, b-1, \quad t' = 1, 2, \dots, b-t.$$

Peluang kondisi diatas dapat ditentukan berdasarkan distribusi impiris angka random R.

Lalu, kita turunkan persamaan optimal dari periode *interior* dan *boundary*. Program dinamis secara rekursif memformulasikan menjadi

$$V_{kt+i}(x) = \min_{y+x>0, y \geq 0} \{c(y) + EV_{kt+i+1}(x + y - D)\},$$

$$i = 1, 2, \dots, k-1, \quad b = 1, 2, \dots, b-1, \quad (1)$$

disini, $c(y)$ adalah biaya yang dibutuhkan pada periode interior $kt + i$ jika *Jeppesen* memesan sejumlah barang dengan nilai y , dan $EV_{kt+i+1}(x + y - D)$ adalah biaya yang diharapkan dari $kt + i + 1$, dengan asumsi dapat optimal pada masa depan. masalah pada periode $kt+1$ adalah memilih y yang dapat meminimalisir jumlah biaya yang dibutuhkan pada saat ini dan yang akan datang. Sehingga perumusan atau *constraint* jumlah barang yang dipesan (y) pada perumusan (1) berarti perusahaan tidak pernah bersentuhan tangan dengan inventaris pada tengah siklus, dan ketika inventaris nol, maka harus segera dipesan kembali.

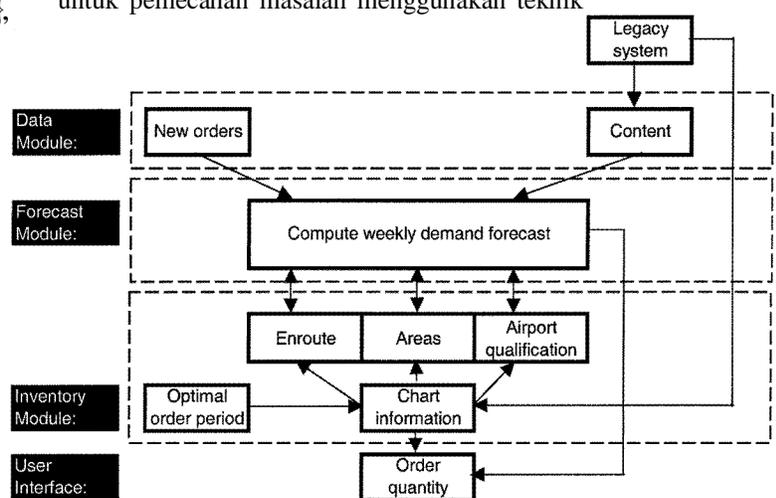
Kita definisikan $[a]^+ = \max \{0, a\}$. Proses rekursif program dinamis mengkoers-pondensikan kedalam periode *boundary* (periode akhir siklus) dengan

$$V_{k(t+1)}(x) = \min_{y+x>0, y \geq 0} \left\{ \begin{array}{l} c(y) + P(R_{kt} > k)EV_{k(t+1)+1} \\ (x + y - D) + P(R_{kt} = k) \\ (AP(D > x + y)) + cE[D - x - y]^+ \end{array} \right\}$$

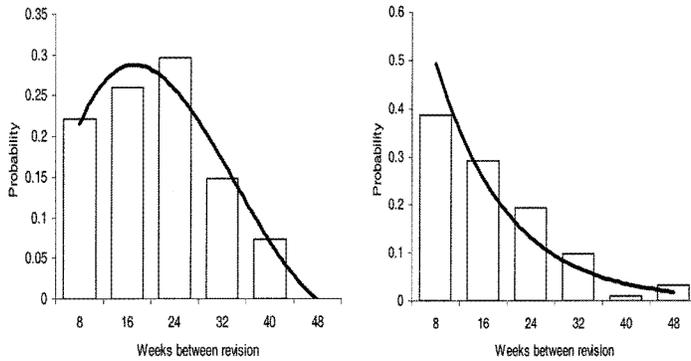
$$t = 0, 1, \dots, b-1. \quad (2)$$

Persamaan optimal diatas dapat dipahami sebagai berikut: $c(y)$ adalah biaya yang terjadi pada periode *boundary* $k(t + 1)$. Dan biaya kedepan tergantung pada hasil pengecekan barang pada akhir periode. Dengan peluang $P(R_{kt} > k)$, hasil pengecekan dengan usia kt akan minimal bertahan pada siklus lainnya, dan proses setelah periode *interior* $k(t+1)+1$ dengan minimum perkiraan biaya $EV_{k(t+1)+1}(x + y - D)$. Dengan probabilitas atau peluang $P(R_{kt}=k)$, dan pengecekan berakhir ($t+1$) yang pertama. Dengan peluang $P(D > x + y)$ dan biaya produksi $cE[D - x - y]^+$. Ketika $t = b-1$ pada persamaan (2), $V_{kb}(x)$ memberikan kondisi *boundary* dari proses rekursif program dinamis dengan $P(R_{k(b-1)} = k) = 1$.

Secara teori, persamaan (1) dan (2) digunakan untuk pemecahan masalah menggunakan teknik



Gambar 1. Proses Pengecekan Inventaris Perusahaan



Gambar 2. Hasil Pencetakan Pengecekan Inventaris Perusahaan

2.4 Keuntungan Yang Diperoleh

Dengan mengimplementasikan sistem pengecekan berbasis program dinamis ini *jeppesen* berhasil mengurangi biaya operasi produksi sampai sebesar 800.00 USD per tahunnya. Pada bulan Juni dan desember 1998 *Jeppesen* mengeluarkan biaya sebesar 250.000 USD untuk membangun sistem pengecekan ini. Pada bulan Januari 1999 berhasil memangkas biaya operasi produksi sampai dengan 500.000 USD pada laporan tahunannya. Dan pada akhirnya sistem pengecekan berbasis program dinamis ini mampu memangkas biaya pada tahun-tahun berikutnya sampai sebesar 800.000 USD.

3. KESIMPULAN

Prinsip dunia industri adalah bagaimana mengoptimalkan output yang dihasilkan suatu perusahaan dengan biaya yang minimum. Sejalan dengan prinsip ekonomi yaitu mendapatkan untung yang maksimum dengan pengorbanan yang seminimumnya. Maka pengelolaan barang atau inventaris perusahaan dapat menjadi salah satu solusi bagaimana meminimumkan biaya yang dikeluarkan oleh perusahaan. Pada makalah ini akan dibahas pengecekan inventaris perusahaan dengan menggunakan pendekatan program dinamis atau *dynamic programming*. Dimana pada program dinamis terdapat karakteristik rekursif, dimana proses rekursif ini mengoptimalkan atau meminimumkan biaya yang dikeluarkan setiap tahap atau *state*-nya dalam pengelolaan pengecekan inventaris perusahaan.

Perusahaan *Jeppesen Sanderson, Inc.* Perusahaan yang bergerak dibidang penyedia produk-produk yang berkaitan dengan teknologi informasi mengimplementasikan manajemen pengelolaan inventarisnya dengan pendekatan program dinamis sehingga mampu memangkas pengeluarannya sampai 800.000 USD tiap tahunnya mulai dari tahun 1998..

REFERENSI

- [1] Elena Katok, dkk, "Jeppesen Uses a Dynamic-Programming-Based DSS to Manage Inventory", Volume 1 Copyright _ 2001 INFORMS, 12 halaman
- [2] Reza Rahman Mohammad, dkk., "Penggunaan Metode *Dynamic Programming* Dalam Perencanaan dan Pengendalian Proyek dengan PERT/CPM", Informatika ITB, 2006, 4 halaman
- [3] Rinaldi Munir, "Strategi Algoritmik", Informatika ITB, 2007

