

BAB 3

LANDASAN TEORI

3.1 Pengertian Program Stokastik

Banyak persoalan keputusan yang dapat dimodelkan dengan menggunakan program matematika yang bertujuan menentukan nilai maksimum atau minimum. Keputusan yang dihasilkan akan bergantung kepada kendala yang dibatasi oleh sumber dana, persyaratan minimum dan lain-lain. Keputusan yang dinyatakan oleh variabel dapat berupa bilangan cacah atau tidak negatif. Tujuan dan kendala adalah fungsi dari variabel, dan persoalan data. Sebagai contoh dari persoalan data termasuk biaya perunit, rata-rata produksi, penjualan atau kapasitas. Andaikan keputusan dinyatakan oleh variabel (x_1, x_2, \dots, x_n) . Sebagai contoh dengan x_i menyatakan produksi ke- i dari n produk. Bentuk umum program matematikanya adalah:

$$\text{Min : } f(x_1, x_2, x_3, \dots, x_n), \quad (3.1)$$

$$\text{Kendala : } g_1(x_1, x_2, x_3, \dots, x_n) \leq 0,$$

$$g_2(x_1, x_2, x_3, \dots, x_n) \leq 0,$$

⋮

$$g_m(x_1, x_2, x_3, \dots, x_n) \leq 0,$$

$$x_1, x_2, x_3, \dots, x_n \in X,$$

dimana X adalah himpunan bilangan real tidak negatif. Program stokastik adalah sebuah nama yang menyatakan program matematika yang dapat berupa linear, cacah, cacah campuran, non linear dengan menampilkan elemen stokastik pada data tunggal dan berfrekuensi. Oleh karena itu dapat dinyatakan bahwa:

1. Pada program matematika deterministik, data (koefisien) adalah bilangan-bilangan yang diketahui (tertentu).
2. Pada program stokastik, data (koefisien) merupakan bilangan yang tidak diketahui (tidak pasti) yang disajikan sebagai distribusi peluang.

Program stokastik merupakan program matematika dengan situasi (yang mengandung) ketidakpastian. Program stokastik merupakan program matematika, dimana beberapa data yang termuat pada tujuan atau kendala mengandung ketidakpastian, ketidakpastian biasanya dicirikan oleh distribusi peluang pada parameter. Walaupun ketidakpastian didefinisikan dengan tepat tetapi pada prakteknya diberikan beberapa skenario (hasil yang mungkin dari data) yang spesifik dan distribusi peluang gabungan yang cepat. Hasil-hasil secara umum digambarkan pada elemen $w \in W$. Ketika beberapa data acak, maka penyelesaian dan nilai tujuan optimal untuk masalah optimisasi juga acak.

Ada dua tipe permasalahan program stokastik, yaitu:

- 1 . Model Rekursif (Recorse Models).
- 2 . Model Kendala Berpeluang (Probabilistically Constrained Models).

Suatu cara logis yang diperlukan dalam persoalan adalah membuat sebuah keputusan sekarang dan meminimumkan biaya rata-rata harapan (yang digunakan) sebagai konsekwensi dari keputusan. Paradigma ini dikenal sebagai model *Recorse*. Andaikan x adalah vektor keputusan yang diambil, dan $y(w)$ adalah sebuah vektor keputusan yang menyatakan aksi terbaru atau konsekwensi dari x . Himpunan berbeda yang berisi y akan dipilih dari tiap-tiap hasil yang mungkin dari w . Formulasi dua tahapnya adalah:

$$\text{Min} : f_1(x) + \text{nilai harapan } [f_2(y(w), w)], \quad (3.2)$$

$$\text{Kendala} : g_1(x) \leq 0, \dots, g_m(x) \leq 0,$$

$$h_1(x, y(w)) \leq 0, \text{ untuk setiap } w \in W,$$

$$\vdots$$

$$h_k(x, y(w)) \leq 0, \text{ untuk setiap } w \in W,$$

$$x \in X \text{ dan } y(w) \in Y.$$

Himpunan kendala h_1, h_2, \dots, h_k menggambarkan hubungan antara keputusan tahap pertama x dan keputusan tahap kedua $y(w)$. Di catat bahwa dipersyaratkan (diharuskan) tiap-tiap kendala dipenuhi dengan peluang 1, atau untuk setiap $w \in W$ yang mungkin. Fungsi f_2 merupakan penyelesaian yang sering muncul dari persoalan matematika. Tidak dibutuhkan untuk membuat korelasi yang berubah-ubah (recorse) untuk keputusan tahap pertama, perlu untuk dibuat korelasi yang terbaik. Model Recorse dapat diperluas dengan banyak cara. Untuk persoalan tahap ganda, pengaruh keputusan sekarang akan ditunggu untuk beberapa ketidakpastian yang diselesaikan kembali (direalisasikan), sehingga pembuatan keputusan yang lain didasarkan pada apa yang terjadi. Tujuannya adalah untuk meminimumkan biaya yang diharapkan dari semua keputusan yang diambil.

Pada beberapa kasus, dapat digunakan suatu metode yang lebih tepat untuk mencoba menentukan sebuah keputusan, yang mana keputusan tersebut dijamin oleh himpunan kendala yang akan dipenuhi oleh sebuah peluang tertentu. Model umum dengan kendala berpeluang disebut sebagai model kendala berpeluang yang dirumuskan sebagai berikut:

$$\text{Min} : f(x_1, x_2, x_3, \dots, x_n), \quad (3.3)$$

$$\begin{aligned} \text{Kendala} : & [g_1(x_1, x_2, x_3, \dots, x_n) \leq 0, \dots, \\ & g_m(x_1, x_2, x_3, \dots, x_n) \leq 0] \geq \alpha, \\ & h_1(x_1, x_2, x_3, \dots, x_n) \leq 0, \\ & h_2(x_1, x_2, x_3, \dots, x_n) \leq 0, \\ & x_1, x_2, x_3, \dots, x_n \in X. \end{aligned}$$

3.2 Program Stokastik Dua Tahap

Banyak persoalan perencanaan dan manajemen yang mengandung resiko dan ketidakpastian dibahas dan diselesaikan dengan program stokastik dua tahap. Persoalan stokastik dengan kompensasi dari divergensi pada sistem dengan kendala mempunyai aplikasi yang lebih banyak dari pada model program yang lain. Penyelesaian persoalan program stokastik dua tahap berisi vektor acak dan vektor deterministik. Pada tahap pertama, penyelesaian persoalan rencana awal deterministik akan dibuat. Pembentukan rencana awal deterministik dilakukan sebelum kondisi acak dari persoalan ditentukan. Sebuah vektor acak pada penyelesaian persoalan yang sesuai digunakan untuk merencanakan kompensasi divergensi, spesifikasi parameter dari persoalan akan muncul pada tahap kedua.

Tujuan dari persoalan di atas adalah meminimum nilai rata-rata biaya, yang mana tidak hanya termasuk pengeluaran pada tahap perencanaan pendahuluan tetapi juga pada tahap kedua yang diperlukan untuk mengkompensasi pada divergensi di dalam sistem kendala persoalan. Jika persoalan program stokastik dengan model dua tahap dapat diselesaikan maka pemilihan dari rencana awal deterministik akan menjamin keberadaan (eksistensi) vektor acak di dalam kompensasi untuk sistem yang divergen.

Andaikan terdapat persoalan berikut:

$$\text{Min} : (C, X), \quad (3.4)$$

$$A^0 X = B^0, \quad (3.5)$$

$$AX = B, \quad (3.6)$$

$$X \geq 0, \quad (3.7)$$

dimana:

$$C = \{c_j\}, j = 1, 2, \dots, m.$$

$$B = (b_i), i = 1, 2, \dots, m.$$

$$B^0 = (b_k^0), k = 1, 2, \dots, m.$$

$$A^0 = \|a_{kj}^0\|, k = 1, 2, \dots, m; j = 1, 2, \dots, n.$$

$$A = \|a_{ij}^0\|, i = 1, 2, \dots, m; j = 1, 2, \dots, n.$$

Andaikan elemen dari matriks $A = A(\omega)$, vektor $B = B(\omega)$ dan $C = C(\omega)$ bernilai acak. Maka untuk proses penyelesaian dari persamaan (3.4 – 3.7) akan dibagi menjadi dua tahapan, sebelum pengamatan dari parameter acak pada kondisi dari tahap pertama dipilih rencana non-negatif deterministik X^0 yang memenuhi kondisi (3.5). Pada tahap kedua ditentukan spesifikasi ω^0 dari setiap kejadian acak yang bersamaan (sesuai) dengan nilai $A(\omega^0)$ dan $B(\omega^0)$. Hitung divergensi $B(\omega^0) - A(\omega^0)X^0$ yang muncul pada kondisi (3.6) setelah realisasi $\omega^0 \in \Omega$. Definisikan vektor kompensasi divergensi $Y \geq 0$ yang sesuai dengan hubungan berikut:

$$D(\omega^0)Y(\omega^0) = B(\omega^0) - A(\omega^0)X^0, \quad (3.8)$$

dimana: $D = \|d_{il}\|, i = 1, 2, \dots, m; l = 1, 2, \dots, n$ adalah sebuah matriks kompensasi yang berisi elemen acak. Sehingga diasumsikan bahwa realisasi acak ω yang diamati pada tahap kedua tidak bergantung pada pemilihan rencana pendahuluan X^0 . Perhatikan persoalan program matematika berikut: Tentukan vektor X

berdimensi n dan vektor $Y(\omega)$ berdimensi $n_1, \omega \in \Omega$. Yang menghasilkan:

$$\min_X E_\omega \{(C(\omega), X) + \min_y (H, Y(\omega))\}, \quad (3.9)$$

dengan kendala:

$$A^0 X = B^0, \quad (3.10)$$

$$A(\omega)X + D(\omega)Y(\omega) = B(\omega), \omega \in \Omega, \quad (3.11)$$

$$X \geq 0, Y \geq 0. \quad (3.12)$$

H adalah vektor akhir yang bergantung pada nilai komponen dari vektor $Y(\omega)$ yang mana merupakan kompensasi divergensi. E adalah notasi ekspekstasi matematika setelah ditentukan rencana awal X^0 , kita pilih komponen vektor $Y(\omega)$ dengan cara menjamin nilai akhir minimum untuk kompensasi divergensi pada kondisi dari persoalan. Dengan kata lain, setelah ditentukan vektor X^0 , perlu menyelesaikan persoalan:

$$\{ \min_Y (H, Y(\omega)) | D(\omega)Y(\omega) = B(\omega) - A(\omega)X^0, Y(\omega) \geq 0 \}. \quad (3.13)$$

Persamaan (3.13) akan mempunyai banyak rencana, vektor $Y(\omega)$ tidak dapat ditentukan pada tiap $\omega \in \Omega$ yang menjamin kondisi (3.11). Persamaan (3.9)-(3.12) dikenal sebagai persoalan program stokastik dua tahap dan persamaan (3.13) adalah persoalan untuk tahap selanjutnya. Model dari penyelesaian persoalan program stokastik (dinamik) dua tahap digunakan untuk perspektif perencanaan dan operasional manajemen, karena selalu terdapat keacakan yang mempengaruhi data yang sudah direncanakan dan sistem manajemen (pelaksanaan). Model dua tahap juga kurang sensitif terhadap perubahan pada parameter dari kondisi persoalan, yang menyebabkan lebih stabil. Akibatnya vektor dapat diterima untuk rencana tahap pertama yang diperlukan untuk setiap $\omega \in \Omega$, terdapat vektor $Y(\omega) \geq 0$ sedemikian hingga:

$$D(\omega)Y(\omega) = B(\omega) - A(\omega)X. \quad (3.14)$$

Andaikan kendala tambahan yang disebutkan secara implisit pada (3.14) muncul pada tahap kedua dari persoalan yang dihasilkan; dan andaikan kondisi yang ditentukan pada vektor non-negatif X dari persamaan (3.10) sudah ditentukan.

Andaikan himpunan $K_1 = \{X : A^0 = B^0, X \geq 0\}$ didefinisikan oleh kendala yang sudah ditentukan tetapi $K_2 = \{X : \forall \omega \in \Omega, \exists Y(\omega) \geq 0, A(\omega)X = B(\omega) - D(\omega)Y(\omega)\}$ didefinisikan oleh kendala yang dihasilkan. Maka himpunan $K = K_1 \cap K_2$ adalah himpunan vektor X yang layak/memenuhi persamaan (3.9)-(3.12). Jika $X \in K$, maka vektor X memenuhi kendala yang sudah ditentukan $A^0 X = B, X \geq 0$ dan disamping itu, persamaan tahap kedua (3.6) akan memiliki banyak rencana. Untuk perhitungan lanjutan diperlukan hasil berikut:

Teorema 3.1. (Kall dan Mayer, 2000) *Himpunan K dengan vektor X pada persoalan program stokastik dua tahap adalah konveks.*

Bukti: $K = K_1 \cap K_2$ tetapi $K_1 = \{X : A^0 = B^0, X \geq 0\}$ adalah konveks. Dengan didefinisikan untuk $\omega \in \Omega$ tertentu (yang ditentukan) himpunan $K_{2\omega} = \{X | \exists Y(\omega) \geq 0\}$ sedemikian hingga $A(\omega)X = B(\omega) - D(\omega)Y(\omega)$ adalah konveks. Hal ini menyatakan bahwa $K_2 = \bigcap_{\omega \in \Omega} K_{2\omega}$ dan $K = K_1 \cap K_2$ adalah himpunan konveks.

3.3 Analisis Persoalan Program Stokastik Dua Tahap

Himpunan K dari rencana pendahuluan untuk persoalan program stokastik dua tahap secara implisit telah ditentukan. Pada kasus generalisasi tidak diketahui bagaimana untuk mengkonstruksi himpunan K_2 . Untuk beberapa kasus parsial, dimana himpunan K_2 mirip dengan R^n . Diasumsikan bahwa rank dari matriks D adalah sama dengan m , kecuali (3.11) dapat disubstitusikan untuk relasi ekuivalen pada p baris, sehingga: $OY = B - AX$, dimana O adalah vektor berisi nol berdimensi m dan p adalah banyaknya baris bergantung pada D . Asumsikan bahwa rank dari matrik D yang berdimensi $m \times n$ adalah sama dengan m dan m kolom pertama adalah linier independen.

Andaikan bahwa untuk tiap $v \in R^n$ terdapat $Y(\omega) \geq 0$ sedemikian hingga:

$$DY(\omega) = v. \quad (3.15)$$

Lemma 3.1.(Kall, 1994) *Jika diasumsikan persamaan(3.15) dipenuhi, maka D mempunyai paling sedikit $(m + 1)$ kolom, yaitu: $n \geq (m + 1)$.*

Teorema 3.2.(Kall, 1994) *Karena sistem persamaan $DY(\omega) = v$ mempunyai penyelesaian non-negatif untuk setiap $v \in R^n$, maka hal ini cukup menunjukkan bahwa terdapat penyelesaian non-negatif untuk sistem homogen dari persamaan linier:*

$$D\pi = 0. \quad (3.16)$$

π lebih besar dari pada 0.

Bukti: Sistem persamaan $DY(\omega) = v$ selalu mempunyai penyelesaian. Mula-mula, andaikan $Y \neq 0$ untuk $j = 1, 2, \dots, m$ tetapi yang lainnya sama dengan nol. Selanjutnya hubungan $\alpha(D\pi) + DY(\omega) = v$ akan dipenuhi untuk sembarang α , jika diambil α yang cukup besar akan diperoleh penyelesaian non-negatif pada persamaan (3.15). Kondisi (3.16) sulit dibuktikan dengan ekspektasi beberapa kasus parsial.

Andaikan n menjadi sama dengan $m = 1$, maka kondisi cukup akan menjadi: $\sum_{j=1}^{m+1} \pi_j D_j$, karena jika $\pi_{m+1} = 0$ akan diperoleh dependen linier dari m kolom pertama D_j , yang mana akan kontradiksi dengan fakta bahwa matriks D mempunyai rank m . Konsekwensinya adalah $\pi_{m+1} > 0$, sehingga diperoleh:

$$-D_{m+1} = \sum_{j=1}^m \pi_j D_j, \pi_j = \frac{\pi_j}{\pi_{m+1}}. \quad (3.17)$$

Sistem di atas adalah persamaan linier yang hanya mempunyai satu penyelesaian. Jika positif maka $K_2 \equiv R^n$. Teorema 3.2 tidak cukup tetapi perlu, sedemikian hingga $DY(\omega) = v$ mempunyai penyelesaian cukup untuk keperluan pemecahan non-negatif dari $DY(\omega) = v$ tidak untuk setiap $v \in R^m$ tetapi hanya untuk

$v = B - AX$ dengan setiap $X \in K_1$ dan setiap $\omega \in \Omega$ sehingga v jauh dari memenuhi untuk semua R^m .

Persamaan (3.4)-(3.7) dapat diinterpretasi sebagai perencanaan produksi, dimana A adalah matriks untuk metode teknologi dasar dan D adalah matriks untuk metode teknologi secara kebetulan pada penentuan varian yang mungkin untuk kompensasi pada divergensi di dalam sistem yang dikondisikan. Pada kasus kondisi dari teorema 3.2 dapat diinterpretasikan dengan cara berikut. Sehingga untuk sembarang divergensi $v \in R^m$, kompensasi Y dapat diterima temuannya, yang dicukupkan oleh metode teknologi kebetulan menyatakan sebuah sistem tertutup, karena itu terdapat intensitas π tidak nol, yang mana semua hasilnya dieksploitasikan oleh metode produksi tertentu dapat dikonsumsi oleh yang lainnya. Sebagai contoh: penjualan dan pembelian dipisahkan dengan baik.

Teorema 3.3. (Kall, 2000) *Karena persamaan (3.13) mempunyai penyelesaian yang berhingga, maka hal ini perlu dan cukup menunjukkan bahwa sistem pertidaksamaan:*

$$ZD \leq H, \quad (3.18)$$

mempunyai penyelesaian.

Pembuktian teorema 3.3 dapat dilihat dengan jelas pada teorema dualitas program linier yang diajukan oleh (Krajewski, 1981). Jika persamaan (3.13) dapat diselesaikan dan mempunyai penyelesaian optimal maka dualnya juga dapat diselesaikan dan begitu juga sebaliknya. Kendala dari persoalan dual untuk (3.13) adalah kondisi (3.17). Kondisi dari teorema 3.3 memiliki kegunaan secara ekonomi. Sehingga biaya pada eksploitasi pada metode teknologi kebetulan dilikuidasi dari divergensi yang berhingga, karena itu cukup dan perlu terdapat sistem estimasi Z untuk menghasilkan metode teknologi kebetulan. Biaya produksi yang disebabkan oleh estimasi output pada metode teknologi yang ke- i tidak lebih tinggi dari eksploitasi dengan singular intensitas dari pada pengeluaran pada eksploitasi dengan singular intensitas.

Teorema 3.4.(Kall dan Mayer, 2000) *Andaikan matriks D mempunyai $m + 1$ kolom dan memenuhi kondisi teorema 3.2 yaitu:*

$$-D_{m+1} = \sum_{j=1}^m \pi_j D_j, \pi_j > 0; \quad j = 1, 2, \dots, m, \quad (3.19)$$

maka untuk pemenuhan kondisi teorema 3.3 adalah syarat perlu dan cukup digantikan relasi (hubungan) berikut:

$$\sum_{j=1}^m \pi_j h_j + h_{m+1} \geq 0, \pi_j > 0, \quad j = 1, 2, \dots, m. \quad (3.20)$$

Bukti: Syarat perlu. Andaikan persamaan tahap kedua (3.13) dapat diselesaikan, maka kumpulan rencana dari masalah dual menjadi tidak kosong. Dengan vektor Z_0 memenuhi kondisi (3.18), maka:

$$Z_0 D_j \leq h_j, \quad j = 1, 2, \dots, m + 1, \quad (3.21)$$

karena itu, dengan $\pi_j > 0$

$$\sum_{j=1}^m \pi_j Z_0 D_j \leq \sum_{j=1}^m \pi_j h_j; \quad Z_0 D_{M+1} = - \sum_{j=1}^m Z_0 \pi_j D_j, \quad (3.22)$$

disamping itu, didapatkan:

$$Z_0 D_{m+1} = - \sum_{j=1}^m Z_0 \pi_j D_j \leq h_{m+1}. \quad (3.23)$$

Dari kondisi (3.20) dan (3.21) diperoleh hasil (3.19) syarat cukup. Andaikan (3.19) digantikan oleh fungsi tujuan pada persamaan (3.13) yang tidak berkendala pada himpunan rencana, maka himpunan rencana persoalan dual untuk persoalan tahap kedua adalah kosong.

$$\{Z | ZD \leq H\} = \emptyset. \quad (3.24)$$

Dari linear independen vektor-vektor D_1, D_2, \dots, D_m jika mengikuti sistem

$$ZD_j = h_j, \quad j = 1, 2, \dots, m, \quad (3.25)$$

hanya mempunyai penyelesaian Z_0 , karena persamaan (3.22) diperoleh

$$Z_0 D_{m+1} > h_{m+1}. \quad (3.26)$$

Dari kondisi teorema dan persamaan (3.23), (3.24) diperoleh

$$Z_0 D_{m+1} = - \sum_{j=1}^m Z_0 \pi_j D_j = - \sum_{j=1}^m \pi_j h_j > h_{m+1}, \quad (3.27)$$

yang mana kontradiksi dengan kondisi (3.18) sehingga teorema dipenuhi.

Kondisi (3.18) menguntungkan secara ekonomi pada persoalan penjadwalan. Andaikan metode teknologi berbentuk sistem tertutup, maka biaya dari eksploitasi metode output yang bertujuan kompensasi divergensi akan berhingga, jika tidak mungkin mendapatkan keuntungan dari rezim yang tidak jalan dari pekerjaan (jika persamaan (3.16) dapat dipenuhi). Dalam pekerjaan yang diajukan oleh (Kall dan Mayer, 2000) ditunjukkan bahwa kondisi analog yang hilang dari keuntungan juga tergantikan dalam kasus ketika $n > m + 1$. Tetapi kondisi ini hanya syarat perlu. Perhatikan sebuah deterministik ekivalen untuk persoalan program stokastik dua tahap pada (3.9)-(3.12) dan tunjukkan bahwa persamaan (3.9)-(3.12) adalah persoalan program konveks. Dual untuk persoalan pada tahap kedua (3.13) adalah:

$$(Z, B - AX) \rightarrow \text{Max}, \quad (3.28)$$

$$ZD \leq H. \quad (3.29)$$

Andaikan penyelesaian persamaan (3.13) ada dan berhingga, maka terdapat penyelesaian berhingga untuk persamaan (3.24)-(3.25) dan nilai optimal untuk keduanya telah dikerjakan oleh (Wallace, 1986). Definisikan nilai fungsi sebagai \emptyset . Dapat diperoleh bahwa $\emptyset(X, A, B)$ menjadi titik maksimum (3.24) yang dicapai dengan kondisi (3.25) untuk X, A, B yang ditetapkan. Sehingga untuk sembarang

X_1 dan X_2 nilai ekstrimum fungsi tujuan (3.24) adalah berhingga, diperoleh:

$$\begin{aligned} & Z(\alpha X_1 + (1 - \alpha)X_2, A, B)(B - A(\alpha X_1 + (1 - \alpha)X_2)) = \\ & Z(\alpha X_1 + (1 - \alpha)X_2, A, B)[\alpha(B - AX_1) + (1 - \alpha)(B - AX_2)] \leq \\ & \alpha Z(X_1, A, B)(B - AX_1) + (1 - \alpha)Z(X_2, A, B)(B - AX_2). \end{aligned}$$

Andaikan \emptyset adalah fungsi tujuan dari persoalan deterministik ekivalen, maka \emptyset adalah fungsi konveks, karena kombinasi non-negatif fungsi konveks adalah fungsi konveks. Dari konveksitas fungsi tujuan \emptyset mengikuti kontinuitas pada setiap titik dalam dari himpunan konveks K . Oleh karena itu dibuktikan oleh pernyataan berikut.

Teorema 3.5. (Kall, 1994) *Deterministik ekivalen untuk persoalan program stokastik dua-tahap (3.9)-(3.12) adalah persoalan program konveks.* Pernyataan selanjutnya memberikan dasar teori untuk mengkonstruksi pendekatan numerik pada penyelesaian persoalan dua-tahap. Perhatikan metode untuk menyelesaikan persoalan dua tahap diperlukan penggunaan hubungan (persamaan) fungsi dasar untuk $\emptyset(X)$ dan menyediakan kondisi differensiabel $\emptyset(X)$. Pada bagian ini akan digunakan fungsi dasar untuk fungsi konveks $F(\mu)$ pada titik $\mu_0 \in M$, untuk fungsi linear L , jika $F(\mu) - F(\mu_0) \geq (L, \mu - \mu_0)$ untuk setiap $\mu_0 \in M$. Hal ini dapat dilihat pada teorema 3.6.

Teorema 3.6. *Fungsi:*

$$E\{C - Z^*(A, B, X_0)A\} = \int_{\Omega} \{C(\omega) - Z^*[A(\omega), B(\omega), X_0]A(\omega)\} dp,$$

adalah dasar untuk fungsi tujuan dari persoalan deterministik ekivalen pada titik $x_0 \in K$. Di dalam pembahasan yang dikerjakan oleh (Kall, 1994) telah dinyatakan bahwa jika ukuran peluang pada ruang A, B kontinu absolut relatif terhadap ukuran lebesgue pada ruang A, B dan kondisi tertentu dipenuhi maka fungsi tujuan $\emptyset(X)$ yang merupakan persoalan deterministik ekivalen adalah kontinu differensiabel setiap tempat pada himpunan K untuk investigasi kondisi optimalitas rencana

X pada persoalan tahap pertama, dibutuhkan vektor $C_x = E[C - Z^*(A, B, X)A]$ dan bentuk linear $L_x = (C_{x_1}, X) = E[C - Z^*(A, B, X)A]$. (Kall, 1994) mengajukan formulasi kondisi perlu untuk optimalitas pada rencana X di dalam persoalan program stokastik dua tahap.

Teorema 3.7. (Kall, 2000) *Jika X^* adalah rencana deterministik untuk persoalan dua tahap maka untuk sembarang $X \in K$,*

$$L_x(X^*) \leq L_x(X). \quad (3.30)$$

Bukti: Andaikan X^* adalah rencana optimal tetapi X rencana yang diterima untuk persoalan dua tahap. Dapat diperoleh : $\emptyset(X^*) \leq \emptyset(X)$

$$E(CX^* + Z^*(A, B, X^*)(B - AX^*)) \leq E(CX + Z^*(A, B, X)(B - AX)), \quad (3.31)$$

$$E(Z^*(A, B, X^*)(B - AX^*)) \geq E(Z^*(A, B, X)(B - AX)). \quad (3.32)$$

Kurangkan (3.28) dari (3.27), dan diambil $Z^*(A, B, X^*)$ sebagai rencana optimal untuk masalah dual dan diperoleh hasil (3.26). Melalui pekerjaan yang diajukan oleh (Kall, 2000) diperoleh bahwa kemungkinan untuk membuat kegunaan secara ekonomi pada kondisi (3.26). Vektor $Z^*(A, B < X)$ adalah penyelesaian masalah dual untuk persoalan dua tahap dan merupakan vektor estimasi untuk produk jarang (kurang) atau berlebihan pada intensitas X dari metode teknologi setelah matriks teknologi A dan vektor permintaan B yang di realisasikan. Estimasi ini mendefinisikan pengaruh dari nilai divergensi pada pengeluaran untuk likuidasi ekonomi dari divergensi. Nilai:

$$\sum_{i=1}^m a_{ij} Z_i^*(A, B, X) - C_j,$$

menunjukkan keuntungan dari eksploitasi pada metode teknologi dengan intensitas singular, dengan perkiraan parameter persoalan direalisasikan sebagai elemen matriks A dan komponen vektor B dan C , tetapi estimasi produk dihitung untuk kasus di dalam eksploitasi metode teknologi yang dikerjakan dengan intensitas X .

Jika vektor X^* mendefinisikan rencana awal optimal untuk persoalan program dua tahap, rekapitulasi keuntungan rata-rata pada intensitas X^* selama penggunaan metode produksi teknologi dihitung pada optimisasi yang tidak lebih kecil dari rekapitulasi keuntungan rata-rata yang dihitung pada estimasi optimal untuk sembarang rencana lain X yang dibolehkan. Akan diformulasi tanpa pembuktian teorema pada kondisi cukup dan perlu dari optimisasi untuk rencana persoalan program stokastik dua tahap.

Teorema 3.8. *Andaikan X^* titik internal dalam dari himpunan K , tetapi sebuah fungsi objektif $\emptyset(X)$ pada persoalan deterministik ekuivalen terhadap persoalan dua tahap yang berbeda di dalam neighbourhood dari titik X^* . Maka persoalan dual $Z^*(A, B, X^*)$ sedemikian hingga:*

$$C_{X^*} = E[C - Z^*(A, B, X^*)A] = 0. \quad (3.33)$$

Jika dan hanya jika X^* adalah penyelesaian persoalan dua tahap (Kall, 2000).

3.4 Program Stokastik Tahap Ganda

Persoalan program stokastik dinamik digeneralisasi oleh kasus dua tahap. Banyak persoalan praktis yang berupa perencanaan, perancangan dan manajemen tidak dapat digambarkan dengan bantuan model statis. Perencanaan dengan periode waktu yang panjang berkembang pada sistem ekonomi, kontrol operasional pada peralatan militer, regulasi pada proses teknologi dan persoalan lain yang termasuk pada parameter acak dan mengharuskan deskripsi untuk penggunaan model probabilistik dinamik. Untuk model bertujuan, metode program stokastik tahap ganda seringkali digunakan. Model program stokastik tahap ganda dan metode untuk realisasi secukupnya bergantung pada informasi mengenai nilai parameter di dalam kondisi persoalan, yang mana memiliki waktu untuk membuat keputusan selanjutnya. Terdapat persoalan dinamik yang mana tiap-tiap tahap berurutan disyaratkan untuk melengkapi kompensasi divergensi yang dikondisikan oleh kon-

disi realisasi persoalan dan oleh pembuatan keputusan tercepat (tahap sebelumnya). Pada masalah yang lain disyaratkan bahwa tiap-tiap tahap peluang yang memenuhi kendala tidak melebihi nilai tertentu yang diberikan sebelumnya atas ekspektasi matematika pada fungsi dari divergensi di dalam kondisi yang dibatasi oleh bilangan yang diberikan atau nilai dari fungsi pada parameter acak yang direalisasikan pada tahap sebelumnya.

Pada bermacam-macam proses aktual yang dapat dimodelkan, menyebabkan persoalan dinamik akan memiliki salah satu bentuk berikut yaitu: tidak dapat dikondisikan, kondisi probabilistik atau kendala statistik. Untuk persoalan dinamik dengan kendala tidak dapat dikondisikan, karakteristik keputusan adalah membuat pada basis informasi mengenai distribusi yang dikombinasikan oleh parameter acak dari kondisi pada semua tahapan. Pada persoalan dinamik dengan kondisi dua kasus kendala dapat dibedakan menjadi: (a) jika oleh momen pembuatan keputusan hanya realisasi dari parameter acak pada tahap sebelumnya yang diperkirakan diketahui dan (b) jika oleh momen pembuatan keputusan melengkapi informasi yang tersedia mengenai realisasi parameter acak (termasuk kondisi) yang dinyatakan tahapan, tetapi nilai dari parameter acak pada tahapan berurutan tidak diketahui. Terdapat relasi tertentu antara persoalan tahap ganda dengan yang tidak dapat dikondisikan dan kondisi kendala.

Penyelesaian optimal untuk persoalan program stokastik dinamik diperoleh dengan strategi murni atau campuran. Pada komponen kasus akhir penyelesaian atau karakteristik statistik dari distribusi yang memberikan penyelesaian akan bergantung pada nilai parameter acak di dalam kondisi persoalan yang telah direalisasikan oleh momen pembuatan keputusan. Konstruksi model probabilistik dinamik dan pelaksanaan metode untuk realisasi yang ditampilkan akan sangat sulit. Pada bagian ini akan diberikan beberapa persoalan yang berisi model matematika untuk persoalan tahap ganda dan prosedur untuk mengkonstruksi penyelesaiannya.

Untuk perhitungan selanjutnya dan analisis persoalan program stokastik tahap ganda, didefinisikan konsep yang diberikan berikut ini. Andaikan terdapat tahap ke- i yaitu $\Omega_i, i = 0, 1, \dots, n$ untuk beberapa ruang kejadian elementer ω_i , dimana Ω_0 berisi satu elemen ω_0 . Andaikan Ω^k adalah hasil khusus dari $\Omega_i, i = 1, 2, \dots, k; \omega^k = (\omega_1, \dots, \omega_k), \Omega^n = \Omega$ dan andaikan pada Ω diberikan ukuran probabilistik p yang didefinisikan dengan cara: jika $A \subset \Omega^k$ maka $p^k(A) = P(A \times \Omega^{k+1} \times \dots \times \Omega^n)$. Diperkenalkan ruang probabilistik (Ω, Σ, P) dengan Σ berkaitan dengan σ -aljabar, didefinisikan p_k sebagai kondisi ukuran probabilistik pada Ω^k .

$$P_k(A|\omega^{k-1} \in B) = \frac{P_k(A \times B)}{P_k(\Omega_k \times B)}$$

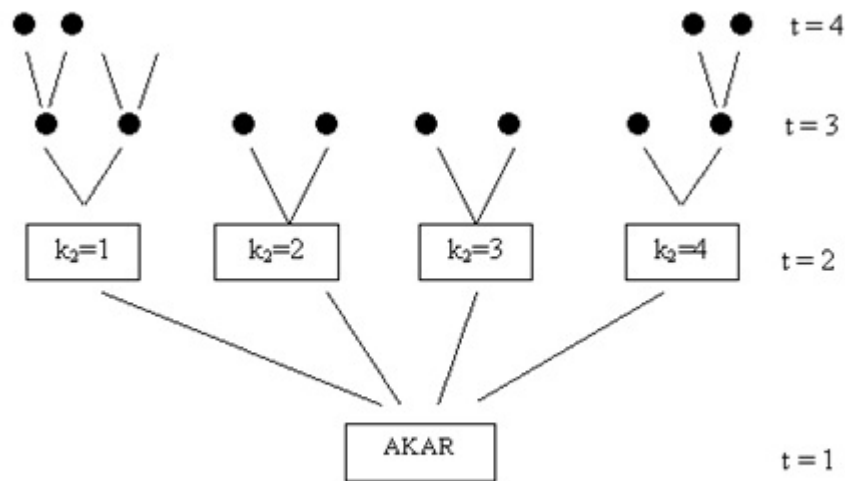
untuk sembarang $A \subset \Omega_k, B \subset \Omega_{k-1}$.

X^k disebut sebagai descartian product $X_i, i = 1, 2, \dots, k; X^k = (x_1, \dots, x_k) \in X^k, X^n \equiv X$ dimana X_0, X_1, X_n adalah barisan dari himpunan struktur sembarang $X^k \in X^k, k = 0, 1, \dots, n$ dan himpunan X termasuk satu titik X_0 . Andaikan m_k diberikan sebagai fungsi vektor pada $\varphi_k(\omega^k, X^k)$ berdimensi untuk setiap $\omega^k \in \Omega^k, X^k \in X^k, k = 1, 2, \dots, n$ dan juga untuk setiap $\omega \in \Omega$ pada himpunan X fungsi $\varphi_k(\omega^k, X^k)$. Masukkan himpunan acak $G_k^0 = G_k^0(\omega^k)$ dan $b_k(\omega^{k-1})m_k$ adalah fungsi vektor. B_k berdimensi acak dari ω^{k-1} (dibatasi dan terukur) dinyatakan sebagai ruang Banach yang termasuk pada fungsi vektor berdimensi, dapat ditulis $b_k(\omega^{k-1}) \sum_{i=1}^k m_i$. Akhirnya, $E_{\omega^k}(U(\omega^k)|\omega^{k-1})$ menyatakan kondisi ekspektasi matematika $U(\omega^k)$ dibawah perkiraan realisasi ω^{k-1} yang diketahui.

3.5 Pengertian Pembentukan Pohon Skenario

Dalam banyak aplikasi, sebaran peubah acak tidak diketahui atau walaupun diketahui, terlalu mahal untuk memperhatikan sebaran diskrit dengan banyak hasil yang mungkin atau menangani sebaran kontinu dengan integrasi numerik. Merupakan hal yang umum untuk memilih himpunan hasil representatif yang relatif kecil yang disebut skenario untuk menyajikan kejadian acak. Skenario merupakan

kuartil dari sebaran yang diketahui atau data nyata, prediksi dan beberapa pohon atau dibangun dengan simulasi. Setiap skenario diberikan nilai probabilitas untuk merefleksikan kemungkinan kejadiannya. Untuk model tahap ganda, informasi skenario dapat diorganisasikan ke dalam struktur pohon. Gambar 1, memberikan contoh pohon skenario untuk persoalan 4 tahap. Buhul akar menyatakan



Gambar 1. Pohon Skenario

waktu sekarang atau bagian dari data yang diketahui. Pada tahap 2, terdapat 4 kemungkinan berbeda dan setiap dari padanya mempunyai berbagai hasil berbeda yang mungkin di tahap 3 dan seterusnya. Suatu skenario terdiri dari lintasan lengkap dari buhul akar ke satu buhul daun, mendefinisikan realisasi tunggal dari himpunan peubah acak. Ambil jumlah tahap T dan jumlah hasil yang mungkin dalam setiap tahap dapat dilabel secara berurutan oleh K_t , untuk $t = 1, \dots, T$. Buhul di setiap tahap dapat dilabel secara berurutan dengan $k_t = 1, \dots, K_t$ untuk semua t . $D_t(k)$ menyatakan turunan langsung dalam waktu t dari buhul k . Misalnya dalam pohon skenario di Gambar 1. D3 (1) memperlihatkan turunan langsung dari buhul 1 yang merupakan dua buhul paling kiri dalam waktu 3. Untuk setiap buhul daun k dalam tahap T , andaikan P_t^k merupakan probabilitas

terkait dari keterjadian skenario. Untuk $t = T - 1, \dots, 1$. Dengan P_t^k diberikan oleh :

$$p_{t+1}^k = \sum_{l \in D_{t+1}} p_{t+1}^l$$

dengan $p_1 = 1$ Pohon keputusan memberikan kelenturan kepada pemodel untuk memilih skenario yang diperlukan untuk diperhatikan dan kepentingannya. Begitupun tidaklah praktis untuk memperhatikan terlalu banyak skenario. Ini terutama terjadi untuk persoalan dimana banyak mengandung faktor acak.

BAB 4

PEMBAHASAN DAN HASIL

4.1 Sifat Eksogen Proses Acak

Proses acak dikatakan eksogen karena memungkinkan untuk mensimulasi, memilih dan mengatur terhadap realisasi kemungkinan proses eksogen. Sebelum pengamatan dilakukan, dan mengoptimalkan secara bersama-sama keputusan dengan kenyataan yang mungkin. Dalam hal untuk memisahkan deskripsi ketidakpastian dan optimalisasi keputusan yang mungkin muncul pada awalnya, yang dapat dimodelkan dengan teknik pemrograman stokastik.

Dalam hal ini untuk masalah teori kontrol dimana ketidakpastian diidentifikasi untuk beberapa pengamatan atau sistem dinamik, dan ketidakpastian dipedomani pada nilai parameter sistem, di sisi lain apa yang dapat dilihat dari sekuensial pengambilan keputusan masalah di bawah ketidakpastian. Sumber utama ketidakpastian adalah justru orang-orang yang kurang dipengaruhi perilaku dari para pengambil keputusan (suku bunga evolusi, peraturan baru), dan proses acak sangat dipengaruhi oleh perilaku para pengambil keputusan dalam proses keputusan awal dan memperlakukannya sebagai proses keputusan.

Probabilistik didasarkan pada himpunan bagian dari pohon skenario yang mungkin dapat dilihat dari proses acak berlawanan yang akan mengeksploitasi salah satu skenario selama proses perencanaan. Dalam berbagai masalah dan situasi, lingkungan merupakan kendala yang mungkin untuk memilih ukuran kinerja yang kuat untuk hasil yang buruk, tetapi keadaan tersebut dapat dioptimalkan. Model proses eksogen adalah kemungkinan dari pengamatan seperti untuk time series, sedangkan pembelajaran model untuk proses endogen dibahas untuk dapat mensimulasikan keadaan transisi yang mungkin untuk setiap tindakan, atau

setidaknya memiliki kumpulan data yang lengkap dari tindakan yang berkaitan dengan transisi kelompok. Penjadwalan unit tenaga listrik (Frantzeskakis dan Powell, 1989) dan manajemen arus kas, aset dan kewajiban (Dempster et.al, 2008) adalah contoh (keputusan berurutan membuat masalah dengan proses eksogen mengikuti model yang canggih).

4.2 Perbandingan Proses Keputusan Markov

Dalam Proses Keputusan Markov pengambil keputusan berusaha untuk dapat mengoptimalkan kriteria kinerja menjadi sebuah solusi. Hal itu merupakan bagian dari pengambil keputusan pada waktu t yang bertepatan dengan penyederhanaan dalam sistem dinamik, sehingga tidak dapat dipertimbangkan dalam diskusi parsial atau risiko-sensitivitas, dimana keadaan sistem tidak dipengaruhi oleh keadaan informasi dari pembuat keputusan. Kebijakan mengambil keputusan yang optimal sering ditemukan dalam pemrograman dinamik, namun masalah pemrograman stokastik multistage pemrograman dapat dipandang sebagai himpunan bagian dari proses keputusan Markov, dengan mengidentifikasi hasil yang tumbuh dari pengamatan $(\xi_1, \dots, \xi_{t-1})$ sekelompok yang membuat keputusan.

Formulasi pemrograman stokastik pada kenyataannya sangat berbeda, hasil kompleksitas menunjukkan bahwa algoritma yang dipakai sangat efisien ketika ruang keputusan terbatas dan kecil ((Mulvey, 1989), (Mulvey, 1991)). Sedangkan untuk pohon skenario berdasarkan kerangka program stokastik, resolusi efisien ketika masalah optimasi adalah cembung, khususnya ruang keputusan kontinu, dan jumlah tahap keputusan kecil (Kouwenberg, 2001). Salah satu teknik pemrograman stokastik adalah kemampuan untuk menangani secara efisien dengan ruang dimensi keputusan yang berkelanjutan, terstruktur oleh banyak kendala, dan dengan pasti dalam proses acak. Pada saat yang sama, jika model pemrograman stokastik telah digunakan untuk mengoptimalkan keputusan jangka panjang yang dilakukan sekali dan memiliki konsekuensi yang baik, misalnya dalam

perencanaan kapasitas jaringan, yang sekarang semakin banyak digunakan dalam konteks optimal dan strategi pengendalian yang terbatas.

Dalam penggunaan ini, pada setiap tahap keputusan model yang diperbarui selama perencanaan sisanya dapat dibangun kembali dan dioptimalkan dari tahap pertama keputusan yang benar-benar diterapkan. Sehingga, ketika sebuah program stokastik diselesaikan pada pohon skenario, dapat dicari kebijakan keputusan yang mengarah ke urutan keputusan relatif yang didapat dari pohon skenario. Keputusan tahap pertama tidak tergantung pada pengamatan dan dengan demikian selalu diterapkan pada setiap skenario baru, sedangkan keputusan recourse relatif terhadap pohon skenario tertentu dan tepat pada skenario baru, terutama jika himpunan kelayakan tergantung pada proses acak.

4.3 Curse Dimensi

Curse Dimensi merupakan algoritma kompleksitas komputasi untuk memilih kebijakan optimalnya lebih tinggi. Ruang masukan dimensi memerlukan eksponensial pertumbuhan sumber daya komputasi yang mengarah ke formulasi masalah yang akan diselesaikan. Dalam pemrograman dinamik, ruang input adalah ruang keadaan. Dalam prakteknya curse merupakan batas dimensi yang mencakup masukan ke dalam R^d . Perkiraan Pemrograman Metode Dinamis (PPMD) (Buchholz, 1995) dan pendekatan penguatan (Bunke dan Caelli, 2001) membantu untuk mengurangi curse. Misalnya dengan mencoba untuk menutupi daerah ruang masukan yang benar-benar mencapai di bawah kebijakan yang optimal.

Komponen eksplorasi dapat ditambahkan ke pemrograman dinamik dengan strategi solusi yang ada sehingga ditemukan daerah yang layak dari ruang input oleh pembuat keputusan. Beberapa pendekatan yang ada pada struktur kebijakan yang dekat-optimal. Sebagai contoh, ketika kebijakan dekat-optimal dapat dijelaskan oleh sejumlah kecil parameter, ada satu himpunan terbatas dan ni-

lainnya kecil dari tindakan, yang dikenal dengan priori, yang merupakan dasar dari kebijakan dekat-optimal, dan digunakan untuk menggerakkan tahap eksplorasi. Situasi seperti ini sering dimanfaatkan dalam robotika.

Struktur dari masalah optimasi adalah sebuah keputusan dan ruang input yang diidentifikasi pada fase awal eksplorasi sesuai dengan kebijakan dekat-optimal. Ini menjamin keberhasilan dari strategi eksplorasi yang optimal, dalam hal ini bahwa untuk memperbaiki keputusan dalam ruang lingkup yang menjanjikan. Situasi ini biasanya muncul dalam masalah dimana bagian stokastik berasal dari proses yang sedikit mengganggu dinamika sistem. Algoritma pemrograman stokastik tidak bergantung pada ruang pemrograman dinamik, sebaliknya bergantung pada serangkaian sifat eksogen proses acak yang tidak sesuai dengan ruang keadaan (formula tambahan x_t diperlakukan dalam contoh bagian sebelumnya).

Hal tersebut dilengkapi dari ruang keputusan yang "dieksplorasi" selama optimasi prosedur itu sendiri. Keberhasilan pendekatan dengan demikian akan bergantung pada optimisasi ruang, dan bukan pada wawasan tentang struktur dekat-optimal kebijakan. Dalam pendekatan multistage pemrograman stokastik, curse dimensi ada ketika jumlah tahap keputusan meningkat, oleh karena itu, metode tersebut bisa digunakan untuk analogi tertentu yang berhubungan dengan pemrograman stokastik. Metode perkiraan program stokastik yang hanya mencakup realisasi dari sifat eksogen proses acak dan dipakai untuk mendapatkan dekat-optimal keputusan. Metode ini bergantung pada sejumlah skenario yang tidak tumbuh dengan pesat dengan dimensi proses eksogen dan jumlah tahapan.

4.4 Pemrograman Stokastik Multi-Tahap

Dengan curse dimensi, pemrograman stokastik multi-tahap memiliki model dengan keputusan tertentu. Pada saat yang sama terdapat kerangka gabungan antara keputusan yang disederhanakan. Pengurangan pengendalian model prediktif dalam penyederhanaan secara radikal dapat menyeleksi informasi probabilistik rinci tentang ketidakpastian, mengambil skenario nominal, dan mengoptimalkan keputusan pada skenario nominal. Sebagai prakteknya untuk mendefinisikan skenario nominal yaitu dengan menggantikan variabel acak oleh ekspektasi yang ada, dan masalah yang dihasilkan pada skenario nominal disebut masalah nilai yang diharapkan, serta solusi yang merupakan rencana nominal. Jika rencana nominal dapat digunakan sebagai kebijakan keputusan, maka pengambil keputusan biasanya ingin mengambil ulang rencana pada tahap keputusan berikutnya dengan memperbaharui masalah yang diharapkan pada nilai skenario nominal dengan menggabungkannya ke pengamatan. Dalam komunitas kontrol, pendekatan ini dikenal sebagai Model Prediktif Kontrol (MPK). Nilai keputusan pemrograman multistage dengan model keputusan kontrol prediktif diberikan oleh nilai dari Solusi Stokastik (SS). Untuk membuat gagasan yang tepat, dapat didefinisikan sebagai berikut:

1. V^* , nilai optimal pemrograman stokastik multi-tahap $\min_{\pi} E\{f(\xi, \pi(\xi))\}$. Untuk notasi sederhana, dapat dinyatakan $f(\xi, \pi(\xi)) = 1$ jika kebijakan π adalah keputusan yang tidak layak.
2. $\xi = (\xi_1, \dots, \xi_T)$ skenario nominal
3. u^ξ skenario nominal solusi optimal untuk masalah nilai yang diharapkan $\min_u f(\xi, u)$, dengan catatan bahwa optimasi pada akhir urutan tunggal pada keputusan layak, yang data masalah ditentukan oleh ξ .
4. u_1^ξ dimulai dari keputusan tahap pertama oleh u^ξ

5. V^ξ , nilai optimal pemrograman stokastik multi-tahap $\min_{\pi} E\{f(\xi, \pi(\xi))\}$ ditunjukkan pada kendala tambahan $\pi_1(\xi) = u_1^\xi$ untuk setiap ξ . Sedemikian hingga, dapat ditulis π_1 sebagai variabel optimasi. Untuk nilai dari fungsi konstan bernilai π_1 dengan kendala tambahan hanya $\pi_1 = u_1^\xi$ sesuai definisi V^ξ adalah nilai dari kebijakan pelaksanaan keputusan pertama dari masalah nilai yang diharapkan, dan mengarah ke tahap keputusan yang optimal untuk tahap keputusan selanjutnya. Keputusan tahap berbeda secara umum dari keputusan yang akan dipilih untuk mendapatkan kebijakan optimal dalam program stokastik multi-tahap.

SS kemudian didefinisikan sebagai perbedaan $V^\xi - V^* \geq 0$. Untuk memaksimalkan masalah tersebut, akan ditentukan oleh $V^* - V^\xi \geq 0$. (Lechuga, 2006) menggambarkan kasus khusus (dengan dua tahap keputusan, dan pembatasan keacakan yang mempengaruhi masalah data) yang dimungkinkan untuk menghitung batas pada SS. Pada tahap kesimpulan dari perhitungan survei mempelajari SS, bahwa tidak ada aturan yang dapat memprediksi secara prioritas apakah SS rendah atau tinggi untuk kasus yang diberikan. Misalnya meningkatkan varians variabel acak dapat meningkatkan atau mengurangi SS.

Reduksi dua-tahap pemrograman stokastik adalah sebuah penyederhanaan yang kurang radikal sehingga menyebabkan perbedaan antara tahap recourse, dalam model tahap pertama (terkait dengan ketidakpastian penuh) dan tahap kedua (terkait dengan skenario yang direalisasikan). Sebuah model multi-tahap berdegenerasi menjadi model dua-tahap ketika pohon skenario memiliki cabang hanya pada satu tahap (telah menggambarkan bagaimana variabel acak dan keputusan harus digabung jika pengamatan dan keputusan tidak terhubung). Situasi ini muncul ketika skenario sampel pada data sebelumnya terpisah, yaitu suatu pohon memilikinya, maka cabang hanya pada akar (Kall, 2000). Pendefinisikan Nilai Pemrograman Multi-tahap (NPM) sebagai perbedaan nilai optimal model multistage dibandingkan model dua tahap. Dengan menetapkan batas pada NPM

dan menggambarkan aplikasi (dalam industri semikonduktor) dimana NPM tinggi. Namun perlu dicatat bahwa generalisasi dari SS bukan menghitung bagaimana keputusan multi-tahap mengungguli dua-tahap keputusan, melainkan hal tersebut diimplementasikan dengan membangun kembali model pada setiap tahap, dengan cara skema MPK.

Pengurangan heuristik berdasarkan optimasi parametrik dalam halnya sebagai penyederhanaan antara masalah nilai yang diharapkan dan pengurangan ke model dua tahap, dalam hal ini untuk mengoptimalkan urutan keputusan secara terpisah pada masing-masing skenario. Para pengambil keputusan dapat menggunakan beberapa strategi pemilihan untuk melaksanakan keputusan tahap pertama dan disimpulkan, kemudian diperoleh nilai akhir tahap pertama keputusan. Dalam hal ini, model harus dibangun kembali dengan skenario pada setiap tahap keputusan. Masalah komputasi keputusan yang optimal secara terpisah untuk masing-masing skenario adalah dikenal sebagai masalah distribusi. Masalahnya muncul pada definisi yang diharapkan untuk mendapatkan Nilai Informasi Yang Sempurna (NIYS), yang mengkuantifikasi nilai tambahan dalam membuat keputusan tersebut bisa tercapai dengan harapan mampu untuk memprediksi di masa depan. Untuk membuat gagasan yang tepat, V^* dilambangkan sebagai nilai optimal minimum program stokastik multi-tahap $\min_{\pi} E\{f(\xi, \pi(\xi))\}$ dengan kendala π . $V(\xi)$ menunjukkan nilai optimal deterministik dengan program $\min_{\mu} f(\xi, u)$, dan V^A menjadi nilai yang diharapkan dari $V(\xi)$. Perhatikan bahwa V^A nilai optimal dari program $\min_{\pi^A} E\{f(\xi, \pi^A(\xi))\}$ dengan kendala π^A , yang merupakan optimasi *decomposable* dalam bagian masalah skenario. NIYS tersebut kemudian didefinisikan sebagai perbedaan $V^* - V^A \geq 0$.

Untuk masalah maksimisasi, didefinisikan oleh $V^* - V^A \geq 0$. Intuitif, NIYS tinggi ketika harus menunda kelayakan terhadap himpunan, karena kurangnya hasil informasi di hasil akhir. NIYS ini biasanya ditafsirkan sebagai nilai pengambil keputusan untuk mengetahui masa depan (Meyn et.al, 2009). NIYS juga me-

nunjukkan nilai ketergantungan urutan keputusan yang terdapat pada skenario tertentu yang dapat dioptimalkan. (Meyn et.al, 2009) menunjukkan pada contoh dengan NIYS rendah dan bagaimana strategi didasarkan pada agregasi keputusan yang dioptimalkan secara terpisah pada skenario deterministik yang memiliki hasil yang minimum, sehingga jika NIYS rendah, heuristik kebijakan terhadap keputusan berkinerja buruk. Ini berarti bahwa pendekatan tidak dapat dilakukan dengan baik dalam prakteknya (Chan dan Ching, 2005). Dalam minimisasi strategi pada serangkaian masalah kombinatorial stokastik yang dalam arti luas sulit untuk dipecahkan pada skenario tunggal, pohon skenario dapat membangun solusi referensi dan kelayakan secara tepat untuk mempercepat optimasi pada skenario baru.

Pada Algoritma Hedging Progresif(AHP), nilai algoritma progresif (Rockafellar, 1995) adalah metode yang menghitung solusi untuk program stokastik multi-tahap pada pohon skenario dengan memecahkan bagian masalah yang berulang kali terpisah pada pohon skenario. Dengan tahap awal keputusan digabungkan dengan non-anticipativity kendala dan diperoleh dengan menjumlahkan keputusan skenario yang ada, dalam ruang lingkup heuristik didasarkan pada masalah distribusi yang disajikan di atas. Algoritma AHP memodifikasi bagian masalah skenario pada setiap iterasi untuk membuat keputusan yang digabungkan oleh non-anticipativity kendala dan keputusan yang optimal. Iterasi tahap pertama yang dilakukan dari keputusan oleh strategi agregasi yang terdapat dalam pohon skenario. Oleh karena itu, AHP menunjukkan bahwa terdapat transisi dasar antara konseptual keputusan yang didasarkan pada masalah distribusi dan model keputusan berdasarkan pada masalah pemrograman stokastik multi-tahap.

4.5 Contoh

Penggambaran perhitungan SS dan NIYS pada program stokastik multi-tahap dengan parameter numerik yang dipilih sedemikian hingga dengan model multi-

stage sangat mempunyai nilai tertentu. Dengan demikian disajikan pengambilan keputusan skema dengan menampilkan tahap pertama keputusan yang suboptimal. Jika keputusan dilaksanakan, dan selanjutnya nilai keputusan terbaik dapat diperoleh, dan nilai tujuan dari kelayakan sangat mempengaruhi nilai suboptimal. Dari w_1, w_2, w_3 dengan bobot $\{+1, -1\}$. Untuk $\xi = (\xi_1, \xi_2, \xi_3)$ menjadi random walk sehingga $\xi_1 = w_1, \xi_2 = w_1 + w_2, \xi_3 = w_1 + w_2 + w_3$. Diperoleh 8 hasil dari x_i yang membentuk pohon skenario dengan kendala tertentu. Proses pengambilan keputusan $u = (u_1, u_2, u_3)$ dengan $u_2 \in R$ dan $u_t = (u_{t1}, u_{t2}) \in R^1$ untuk $t = 1, 2, 3$. Kemudian didapat program stokastik multi-tahap (Boris et.al,2009).

Maximize :

$$\frac{1}{8} \sum_{k=1}^8 \{[0.8u_{11}^k - 0.4(u_2^k/2 + u_{31}^k - \xi_3^k)^2] + u_{32}^k \xi_3^k + [1 - u_{11}^k - u_k^k 12]\}$$

kendala :

$$u_{11}^k + u_{12}^k \leq \forall k$$

$$-u_{11}^k \leq u_2^k \leq u_{11}^k \forall k$$

$$-u_{1j}^k \leq u_{3j}^k \leq u_{1j}^k \forall k \quad \text{dan} \quad j = 1, 2$$

$$C1 : u_1^k = u_1^1 \forall k$$

$$C2 : u_2^k = u_2^{k+1} = u_2^{k+2} = u_2^{k+3} \quad \text{untuk} \quad k = 1, 5$$

$$C3 : u_3^k = u_3^{k+1} \quad \text{untuk} \quad k = 1, 3, 5, 7$$

kendala C1, C2, C3, untuk data masalah, menunjukkan variabel optimasi berlebihan yang dapat dihilangkan.

1. Nilai optimal dari program stokastik multi-tahap adalah $V^* = 0,35$ dengan optimal tahap pertama keputusan $u_1^* = (1, 0)$.
2. Masalah nilai yang diharapkan untuk skenario rata-rata $\varsigma = (0, 0, 0)$ diperoleh dengan menetapkan $\xi^k = \varsigma \forall k$ dan menambahkan kendala C2': $u_2^k =$

- $u_2^1 \forall k$ dan C3': $u_3^k = u_3^1 \forall k$, nilai optimal adalah 1 dengan tahap pertama keputusan $u_1^5 = (0, 0)$, ketika kendala setara dan dibuat implisit untuk masalah yang dapat diformulasikan dengan menggunakan 5 variabel optimasi skalar.
3. Suatu relaksasi dua tahap diperoleh dengan kendala C2, C3. Dengan nilai optimal adalah 0,6361 dengan $u_1^k = u_1^{II} = (0, 6111, 0, 3889)$.
 4. Masalah distribusi diperoleh dengan kendala C1, C2, C3. Nilai optimal adalah $V^A = 0,6444$, dua skenario $\xi^1 = (1, 2, 3)$ dan $\xi^8 = (-1, -2, -3)$ untuk tahap pertama keputusan $u_1^1 = u_1^8 = (0, 7778, 0, 2222)$ dan nilai -0,0556. Keenam skenario lain memiliki $u_1^k = (0, 5556, 0, 3578)$ dan nilai 0,8778, $k = 2, \dots, 7$. Perhatikan bahwa secara umum, (i) skenario dengan optimal yang sama pada awal tahap keputusan dan nilai-nilai yang mungkin memiliki nilai yang berbeda pada tahap keputusan, dan (ii) tahap pertama keputusan bisa berbeda untuk semua skenario.
 5. NIYS sama dengan $V^A - V^* = 0,2944$.
 6. Memecahkan program stokastik multi-tahap dengan kendala tambahan $C_1^S : u_1^k = u_1^S \forall k$ menghasilkan batas atas nilai optimal dari skema apapun yang menggunakan keputusan tahap pertama dari masalah nilai yang diharapkan. Nilai ini $V^S = -0.2$.
 7. SS sama dengan $V^* - V^S = 0,55$.
 8. Memecahkan program stokastik multi-tahap dengan kendala tambahan $C^{1II} : u_1^k = u_1^{II} \forall k$ menghasilkan batas atas nilai optimal dari skema yang menggunakan keputusan tahap pertama dari model relaksasi dua tahap. Nilai ini adalah $V^{II} = 0,2431$. Dengan demikian, nilai dari model multi-tahap selama dua tahap (berbeda dari NPM dari Meyn, 2009), setidaknya $V^* - V^{II} = 0,1069$.

Ringkasan nilai optimal dari $V^* = 0,35$ sampai $V^{II} = 0,2431$ (dengan keputusan tahap pertama dari model dua tahap) dan kemudian $V^s = -0.2$ (Dengan keputusan tahap pertama dari model nilai yang diharapkan). Kita juga bisa mempertimbangkan urutan keputusan dari distribusi masalah, dan memeriksa apakah ada strategi yang bisa mengeksploitasi himpunan tahap pertama keputusan untuk mengeluarkan keputusan tahap pertama yang baik (sehubungan dengan pengambilan keputusan skema untuk tahap berikutnya).

1. Seleksi Strategi: Pemecahan program stokastik multi-tahap dengan kendala pada salah satu tahap pertama keputusan dari distribusi masalah menghasilkan hasil sebagai berikut: nilai optimal 0,3056 jika $u_1^k = (0,7778, 0,2222)$ nilai, optimal 0,2167 jika $u_1^k = (0,5556, 0,3578)$. Dalam memecahkan nilai program stokastik multi-tahap untuk mengukur kualitas dari tahap pertama-suboptimal keputusan, strategi pemilihan membutuhkan perkiraan yang baik dari optimal yang berbeda nilai untuk benar-benar output keputusan terbaik.
2. Strategi Konsensus: Hasil dari himpunan dari 8 tahap pertama keputusan akan menjadi keputusan $(0,5556, 0,3578)$ berhubungan dengan skenario 2 sampai 7. Dengan nilai 0,2167, ini ternyata menjadi keputusan terburuk antara $(0,7778, 0,2222)$ dan $(0,5556, 0,3578)$.
3. Strategi Averaging: Keputusan tahap pertama rata-rata dari himpunan 8 tahap pertama keputusan adalah $u_1 = (0,6111, 0,3239)$. Memecahkan program multi-tahap dengan $u_1^k = u^1$ untuk k semua menghasilkan nilai optimal 0,2431.

Hasil terbaik adalah nilai 0,3056 diperoleh dari strategi pemilihan. Catatan bahwa dalam situasi di mana program multi-tahap dan variannya bisa diselesaikan dengan tepat, yaitu dengan pohon skenario mewakili hasil dari proses acak.

BAB 5

KESIMPULAN

Berdasarkan uraian pada bab sebelumnya, maka dapat diambil kesimpulan sebagai berikut: metode-metode yang ada pada pembahasan sebelumnya merupakan suatu aturan Dalam memecahkan nilai program stokastik dengan Markov Chain, dengan strategi pemilihan membutuhkan perkiraan yang baik dari titik optimal yang berbeda nilainya untuk output keputusan terbaik. Di sini dalam situasi tersebut bisa diselesaikan, yaitu dengan metode yang telah dijelaskan mewakili hasil dari proses acak. Metode tersebut meliputi: perkiraan pemograman metode dinamis, pendekatan kebijakan dekat optimal, dan metode perkiraan program stokastik (dalam curse dimensi), serta beberapa metode dalam pemograman stokastik multi-tahap yakni: pendekatan Model Prediktif Kontrol(MPK), metode Algoritma Hedging Progresif(AHP), yang merupakan suatu aturan untuk memecahkan pemograman stokastik multi-tahap dengan tepat dan diperoleh hasil terbaik dari strategi pemilihan.