Jurnal Matematika Integratif. p-ISSN:1412-6184, e-ISSN:2549-903
Vol. 16, No. 1 (2020), pp. 19–28. doi:10.24198/jmi.v16.n1.27112.19-28

Penyelesaian Metode Dekomposisi Benders pada Model Optimisasi *Robust* Masalah *Mixed Integer Linear Programming* Dua Tahap yang melibatkan Variabel *Recourse*

Diah Chaerani[1], Heri Setiawan[2], Alit Kartiwa [1]

[1] Departemen Matematika FMIPA Universitas Padjadjaran
d.chaerani@unpad.ac.id, alit.kartiwa@unpad.ac.id
[2] Alumni Departemen Matematika FMIPA Universitas Padjadjaran
herisetiawan@gmail.com

Abstrak
Dalam makalah ini dibahas penentuan *Robust Counterpart* untuk Masalah Pemrograman Linear Integer Campuran Dua Tahap yang melibatkan Variabel *Recourse* mengggunakan Metode Dekomposisi Bender. Pembahasan dimulai dengan penentuan *Robust Counterpart* (RC) dari Masalah MILP Dua Tahap yang melibatkan Variabel *Recourse* . Kemudian formulasi RC disajikan sebagai Master dan Inner Problem mengikuti kaidah Algoritma Metode Dekomposisi Bender yang membagi masalah RC ke dalam bagian linear yang mudah diselesaikan dan bagian nonlinear atau integer yang sulit diselesaikan. Pembahasan dilengkapi dengan contoh kasus pada masalah optimisasi perhutanan dalam hal penentuan tebang pilih dan pengiriman kayu untuk industri.

Kata kunci: Robust Optimization, MILP, Variabel *Recourse*, Metode Dekomposisi Benders

Abstract
This paper discusses the determination of Robust Counterpart for the Two-Stage Mixed Linear Integer Programming Problem which involves Recourse Variable using the Bender Decomposition Method. The discussion begins with the determination of the Robust Counterpart (RC) of the Two Stage MILP Problem which involves the Recourse Variable. Then the RC formulation is presented as a Master and Inner Problem following the rules of the Bender Decomposition Method algorithm which divides the RC problem into linear parts that are easily resolved and nonlinear parts or integers that are difficult to solve. The discussion is complemented by case examples on the problem of optimizing forestry in terms of determining selective logging and shipping timber to industry.

Keywords: Robust Optimization, MILP, Recourse Variables, Benders Decomposition Method

2000 Mathematics Subject Classification: 90C10, 90C11, 90C15, 90C17
Received: 2020-04-29, Revised: 2020-06-12, Accepted: 2020-06-19.

1. Pendahuluan

Optimisasi *robust* adalah metodologi untuk menangani masalah yang dipengaruhi oleh ketidaktentuan data dan dimana tidak ada distribusi peluang yang memenuhi untuk ketidaktentuan parameter [1]. Dengan optimisasi *robust*, solusi *robust* yang diperoleh dapat membantu pengambil keputusan untuk menghindari kerugian dari ketidaktentuan. Fokus penelitian ini adalah tentang optimisasi *robust untuk masalah Mixed Integer Linear Programming* (MILP) dua tahap dengan variabel *recourse*.

Menurut Billionet [2] optimisasi *robust* MILP dua tahap dengan variabel *recourse* ini dapat dijadikan sebagai alternatif untuk pendekatan pemrograman linear stokastik dengan dua tahap yang diperkenalkan oleh Dantzig [3]. Menurut Dantzig [3], dalam pendekatannya masalah dengan ketidaktentuan data dimodelkan dengan variabel acak, pengambilan keputusan dilakukan dengan dua tahap. Tahap pertama adalah menemukan nilai sebenarnya yang diambil dari variabel acak, kemudian tahap kedua ketidaktentuannya telah diketahui. Dalam pendekatan ini sayangnya ditribusi probabilitas data harus diketahui yang dalam banyak kasus tidak selalu tersedia.

Untuk mengatasi masalah tersebut, diusulkan penyelesaian dengan pendekatan optimisasi *robust*. Pendekatan yang tidak bergantung pada model probabilitas data. Salah satunya Soyster [4] yang mengusulkan model optimisasi linear untuk data dalam satu set konveks. Akan tetapi pendekatan tersebut adalah pendekatan konservatif, solusi optimal yang diperoleh masih jauh dari masalah nominal. Pada tahun 2004 telah diusulkan model optimisasi *robust adjustable* untuk mengatasi masalah konservatif berlebih ini yaitu lihat Ben-Tal [5] yang memperkenalkan pendekatan *robust* dengan dua tahap di bawah ketidaktentuan. Dalam pendekatan ini Ben-Tal [5] mempertimbangkan dua himpunan variabel, yaitu himpunan pertama harus ditentukan sebelum menyelesaikan ketidaktentuan dan yang lainya dapat dihitung setelah ketidaktentuan diselesaikan.

Kemudian ada Aurelie [6] yang mengusulkan pendekatan *two-stage robust* optimisasi dengan *resource* yang tidak melibatkan probabilistik dari ketidaktentuan data, dan mengijinkan pembuat keputusan untuk menyesuaikan tingkat konservatif dari model serta mempertahankan kelineran. Aurelie [6] menyelesaikan ketidaktentuan ruas kanan dalam masalah pemrograman linear dengan *resource*.

Dalam penelitian ini memaparkan masalah linear yang *robust* dengan ketidaktentuan pada sisi bagian kanan (vektor ruas kanan) dengan dua tahap, yang mana variabel tahap pertama *mixed integer* dan variabel tahap kedua kontinu berdasarkan model yang terdapat pada jurnal *2-stage robust MILP with continuous recourse variables* Billionet [2] menggunakan Metode *Benders Decompotition*. Kemudian dilakukan simulasi numerik menggunakan data pada Industri Kehutanan dan *software* Maple digunakan untuk membantu dalam melakukan perhitungan simulasi numerik.

2. Metode Penelitian

Pada sub bagian ini dibahas secara ringkas metode penelitian yang digunakan dalam penelitian ini, antara lain pembahasan mengenai optimisasi *robust* dan cara menyelesaikan *robust counterpart*. Algoritma *Bender's Decomposition* untuk menyelesaikan masalah optimisasi dibahas secara singkat merujuk pada [12].

2.1. **Optimisasi *Robust*.** Pada bagian ini akan diuraikan tentang bagaimana mencari solusi dari ketidaktentuan adalah Optimisasi *Robust*, sehingga dalam penelitian ini menggunakan metode Optimisasi *Robust* [7]. Pada optimisasi *robust*, masalah pemrograman linear dengan ketidaktentuan data didefinisikan sebagai model matematika sebagai berikut:

$$\min_{x} \left\{ c^T x : Ax \leq d \right\} \tag{1}$$

dimana $c \in R^n$, $A \in R^{m \times n}$, $b \in R^m$, dan koefisien (c, A, d) berada dalam himpunan tak tentu U. Asumsi-asumsi dasar dari Optimisasi *Robust* dengan ketidakpastian agar solusi *feasible* [8], yaitu semua variabel keputusan $x \in R^n$ mewakili keputusan *here and now*, pembuat keputusan sepenuhnya bertanggungjawab atas keputusan yang harus dibuat, jika dan hanya jika data sebenarnya telah ditentukan dalam U, dan kendala dari pemrograman linear tak tentu yang dimaksud adalah hard.

Berdasarkan kondisi di atas, pendekatan optimisasi *robust* mengubah masalah tak tentu Persamaan (1) menjadi masalah deterministik tunggal yang disebut dengan *Robust Counterpart* (RC), yaitu:

$$\pi^* = \min_x \left\{ c^T x : Ax \leq d, x \geq 0, \forall (c, A, d) \in U \right\} \tag{2}$$

vektor x^* adalah solusi optimal *robust* jika semua realisasi $\forall (c, A, d) \in U$, x^* *feasible* dan nilai dari fungsi objektif dijamin bernilai paling besar π^*. Koefisien tak tentu pada fungsi objektif menyebabkan fungsi objektif harus dibatasi oleh suatu nilai t, sehingga masalah *Robust Counterpart* ekivalen dengan :

$$\min_{x,t} \left\{ t : c^T x - t \leq 0, a_i^T x - d_i \leq 0; i = 1, ..., m \right\}, \quad (c, a, d) \in U \tag{3}$$

2.2. **Menyelesaikan *Robust Counterpart*.** Asumsikan $c \in R^n$ dan $d \in R^m$ adalah tentu, maka formulasi *robust* dari 1 yaitu *robust counterpart* adalah sebagai berikut [7]:

$$\min_x \left\{ c^T x : A(\zeta) x \leq d, \forall \zeta \in \mathbb{Z} \right\} \Leftrightarrow \min_x \left\{ c^T x : a^T(\zeta) x \leq d, \forall \zeta \in \mathbb{Z} \right\} \tag{4}$$

dimana $\mathbb{Z} \subset R^L$ menunjukan penggunaan himpunan ketidaktentuan tertentu. Solusi $x \in R^L$ merupakan *robust feasible* jika memenuhi semua kendala tak tentu $[A(\zeta) x \leq d]$ untuk semua realisasi dari $\zeta \in \mathbb{Z}$. Didefinisikan parameter ketidaktentuan:

$$a(\zeta) = a = \overline{a} + P\zeta \tag{5}$$

dimana $\overline{a} \in R^n$, $P \in R^{n \times L}$, dan $\overline{a}$ merupakan nilai nominal. Dapat didefinisikan himpunan U:

$$U = \{a | a = \overline{a} + P\zeta, \zeta \in \mathbb{Z}\} \tag{6}$$

Sebuah kendala yang diperoleh dari (4) dengan substitusi parameter ketidaktentuan dapat dimodelkan sebagai berikut:

$$(\overline{a} + P\zeta)^T x \leq d, \forall \zeta \in \mathbb{Z} \tag{7}$$

Selanjutnya, tantangan dalam Optimisasi *Robust* adalah untuk mencari jenis himpunan ketidaktentuan manakah yang dapat diformulasikan menjadi masalah optimisasi *tractable*. Menurut Ben-Tal [8] untuk menganalisis *Robust Counterpart* yang *computationally tractable* dapat dengan menunjukan *Robust Counterpart* dapat dibentuk menjadi *Linear Programming* (LP), *Conic Quadratic Optimization* (CQO), atau *Semi-definite Optimization* (SDO) [11] [10], seperti berikut:

Teorema 1 *Asumsikan himpunan ketidaktentuan U pada (1) diberikan sebagai affine image dari himpunan terbatas $\mathbb{Z} = \{\zeta\} \subset R^N$, dan $\mathbb{Z}$ merupakan:*
(i) Sistem dari kendala pertidaksamaan linear

$$P\zeta \leq p \tag{8}$$

(ii) Sistem dari pertidaksamaan Conic Quadratic

$$\|P_i \zeta - p_i\|_2 \leq q_i^T \zeta - r_i, i = 1, ..., M \tag{9}$$

(iii) Sistem dari pertidaksamaan Matriks Linear

$$P_0 + \sum_{i=1}^{\dim \zeta} \zeta_i P_i \geq 0 \tag{10}$$

Pada kasus (ii) dan (iii) juga diasumsikan bahwa sistem dari kendala yang mendefinisikan U adalah *strictly feasible* maka *robust counterpart* dari (3) adalah ekuivalen dengan:
1. Masalah pemrograman linear pada kasus (i).

2. Masalah *conic quadratic* pada kasus (ii).
3. Masalah semidefinit pada kasus (iii).

2.3. **Menyelesaikan *robust Counterpart* untuk Himpunan *Ellipsoidal Uncertainty*.** Berdasarkan pembahasan yang disajikan oleh Gorissen [7], himpunan *ellipsoidal uncertainty* dapat dinyatakan sebagai berikut:

$$\mathbb{Z} = \{\|\zeta\|_2 \leq 1\} \tag{11}$$

sehingga dapat didefinisikan himpunan U:

$$U = \{a|\overline{a} + P\zeta, \|\zeta\|_2 \leq 1\} \tag{12}$$

Untuk memperoleh formulasi *robust counterpart* dari himpunan *ellipsoidal uncertainty*, maka himpunan *ellipsoidal uncertainty* harus diterapkan pada pertidaksamaan (7) dan akan ekivalen dengan

$$\max_{\zeta:\|\zeta\|_2 \leq 1} (\overline{a} + P\zeta)^T x = \overline{a}^T x + \max_{\zeta:\|\zeta\|_2 \leq 1} (Px)^T \zeta \leq d \tag{13}$$

Perhatikan bahwa untuk suatu x maka nilai terbaik pada kasus terburuk untuk pertidaksamaan di atas tercapai ketika dipilih vektor satuan:

$$\zeta = \frac{P^T x}{\|P^T x\|} \tag{14}$$

sehingga

$$\max_{\zeta:\|\zeta\|_2 \leq 1} \left(P^T x\right)^T \zeta = \frac{\left(P^T x\right)^T \left(P^T x\right)}{\|P^T x\|} = \frac{\left(P^T x\right)^2}{\sqrt{\left(P^T x\right)^2}} = \sqrt{\left(P^T x\right)^2} = \left\|P^T x\right\|_2 \tag{15}$$

Maka diperoleh formulasi *Robust Counterpart* dari Persamaan (13) yang ekivalen dengan pertidaksamaan (7)

$$\overline{a}^T x + \left\|P^T x\right\|_2 \leq d \tag{16}$$

Menurut Ben-Tal [11] dan Chaerani [10], bentuk akhir dari *Robust Counterpart* ini dijamin menjadi masalah yang *computationally tractable.* Jika formulasi *Robust Counterpart* menghasilkan bentuk lain, maka harus menentukan kembali asumsi parameter tak tentu pada model awal.

2.4. **Metode Dekomposisi Bender.** *Bender's Decomposition* adalah landasan sebuah model matematika yang diharuskan mempartisi/membagi ke dalam bagian linear yang mudah diselesaikan dan bagian nonlinear/integer yang sulit diselesaikan [12]. Kemudian digunakan Algoritma *Benders Decomposition.* Menurut [12], Metode Dekomposisi Bender merupakan metode optimisasi untuk menyelesaikan masalah yang memiliki subproblem fisibel.

$$P(x,y) \equiv \min_{x,y} \left\{c^T x + f(y) : Ax + F(y) = b, x \geq 0, y \in Y\right\} \tag{17}$$

dengan $A \in R^{m\times n}$, $x, c \in R^n, b \in R^m, y \in Y \subset R^p$, dalam hal ini $f(y)$ dan $F(y)$ mungkin non linear dan Y bisa saja diskrit ataupun kontinu. Konsep utama dalam algoritma bender adalah mempartisi variabel menjadi dua himpunan yaitu x dan y kemudaian menyelesaikan masalah pada variabel sulit Y.

Berikut ini merupakan langkah pengerjaan Metode Dekomposisi Benders

(1) Pertama, penting untuk mengamati bahwa untuk nilai tetap $y \in Y$ menjadi masalah Pemrograman Linear dalam variabel x. Ini direpresentasikan secara matematis sebagai Sub-masalah Fisible yang dinotasikan sebagai $P(x|y)$. Selanjutnya, diasumsikan bahwa $P(x|y)$ memiliki solusi optimal hingga x untuk setiap $y \in Y$. Sehingga model (17) dapat ditulis kembali menjadi:

$$P(x|y) = \min_{y} \left\{f(y) + \min_{x} \left\{c^T x : Ax = b - F(y), x \geq 0\right\}\right\}. \tag{18}$$

dimana

$$\min_x \left\{c^T x : Ax = b - F(y), x \geq 0\right\}$$

merupakan *inner optimization problem* dan diasumsikan memiliki solusi optimal x untuk setiap $y \in Y$.

(2) Kedua, mencari formulasi masalah dual untuk *inner optimization problem.* Sehingga model (17) dapat ditulis kembali menjadi

$$P_1(u, y) = \min_y \left\{ f(y) + \max_u \left\{(b - F(y))^T u : A^T u \leq c\right\} \right\}. \tag{19}$$

Fungsi pembatas pada inner problem bebas dari variabel y. Solusi optimal dari inner maximation problem adalah finite (tentu) karena asumsi bahwa

$$\min_x \left\{c^T x : Ax = b - F(y), x \geq 0\right\}$$

memiliki solusi optimal x untuk setiap $y \in Y$ dan solusi optimalnya berada pada titik ekstrim $u \in U$. Sehingga model (19) dapat ditulis kembali menjadi:

$$P_2(u, y) = \min_y \left\{ f(y) + \max_u (b - F(y))^T u \right\}. \tag{20}$$

(3) Ketiga, penentuan *full master problem* dengan cara menuliskan kembali model (20) dapat kembali diformulasikan menjadi sebuah masalah minimasi sederhana sebagai berikut.

$$\begin{array}{ll} \min & f(y) + m \\ s.t & (b - F(y))^T u \leq m, u \in U \\ & y \in Y. \end{array} \tag{21}$$

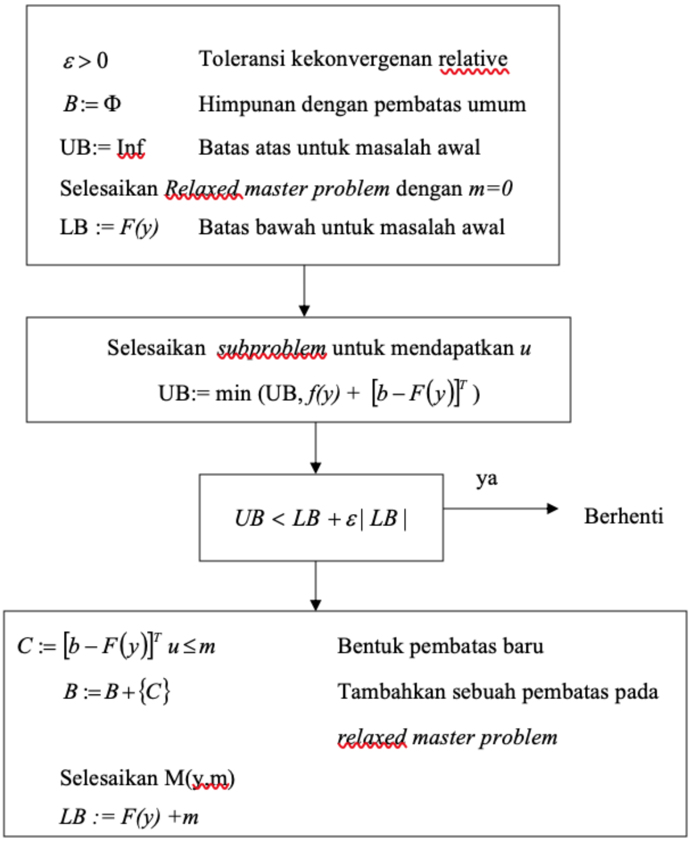
Pada full master problem sangat penting untuk mengetahui bahwa terdapat sebuah kendala untuk masing-masing titik ekstrim. Dalam sebuah masalah bisa saja terdapat jumlah kendala yang sangat besar bahkan pada sebuah masalah yang ukurannya tidak besar. Tetapi hanya sebuah fraksi kecil dari kendala yang akan diikat pada sebuah solusi optimal. Karena itu, model akan diselesaikan secara iterative atau bertahap yang akan dimulai dengan menyelesaikan master problem dengan beberapa atau bahkan tidak ada kendala sementara kedala baru ditambahkan jika dibutuhkan. Dari model (21), maka didefinisikan *relaxed master problem* yaitu:

$$\begin{array}{rll} M(y, m) = & \min & f(y) + m \\ & s.t & (b - F(y))^T u \leq m, u \in B \\ & & y \in Y. \end{array} \tag{22}$$

dimana B merupakan himpunan kosong dan m secara inisial bernilai 0. Bender?s subproblem adalah subproblem yang menyelesaikan sebuah titik ekstrim u dengan sebuah nilai tetap dari $y \in Y$, sehingga dapat ditulis menjadi masalah maksimasi berikut:

$$S(u|y) = \quad \max \quad \left\{(b - F(y))^T u : A^T u \leq c\right\} \tag{23}$$

dengan $u \in R$, $S(u|y)$ memiliki solusi optimal yang hingga dengan asumsi bahwa $P(x|y)$ memiliki solusi optimal yang hingga untuk setiap $y \in Y$. Subproblem diselesaikan untuk mendapatkan nilai u dengan diberikan nilai yang ditentukan dari master problem. Selanjutnya terdapat sebuah uji sederhana untuk menentukan apakah sebuah kendala yang mencakup u harus ditambahkan ke master problem. Jika demkian, maka master problem diselesaikan untuk menghasilkan sebuah nilai baru y sebagai input subproblem yang diselesaikan lagi. Proses ini berlanjut terus sampai dicapai keoptimalan. Langkah-langkah Metode Dekomposisi Bender dapat dilihat pada Gambar 3.



Gambar 1. Diagram Metode Dekomposisi Bender

3. Hasil dan Pembahasan

3.1. Model Optimisasi *Robust* untuk Masalah MILP Dua Tahap dengan Variabel *Recourse* menggunakan Metode *Bender's Decomposition*. Sebelum masuk ke dalam formulasi Optimisasi Robust untuk masalah MILP dua tahap dengan variabel Recourse menggunakan metode Bender's Decomposition, perhatikan model deterministic berikut [2]:

$$\min_{x,y} \left\{ \alpha x + \beta y : \begin{array}{c} Ax + By \geq d \\ Cx \geq b \\ x_i \in N, i = 1, ..., p_1 \\ x_i \in R_+, i = (p_1 + 1), ..., p \\ y \in R_+^q \end{array} \right\} \tag{24}$$

Merujuk Billionet [2], diasumsikan d adalah parameter ketidaktentuan. Sehingga

$$d = \overline{d} + p^T \zeta, \quad \forall \zeta \in \mathbb{Z} \subseteq \mathfrak{D} \tag{25}$$

Maka Persamaan (24) ekivalen dengan

$$\min_{x,y}\left\{\alpha x+\beta y:\begin{array}{c}Ax+By\geq\max_{\zeta\in\mathbb{Z}}\left\{\overline{d}+p^T\zeta\right\}\\ Cx\geq b\\ x_i\in N,i=1,...,p_1\\ x_i\in R_+,i=\left(p_1+1\right),...,p\\ y\in R_+^q\end{array}\right\}\tag{26}$$

Selanjutnya, model (26) dapat kembali diformulasikan menjadi $P_1\left(x,y\right)$:

$$\min_{x}\left\{\alpha x+\min_{y}\left\{\beta y:\begin{array}{c}Ax+By\geq\max_{\zeta\in\mathbb{Z}}\left\{\overline{d}+p^T\zeta\right\}\\ Cx\geq b\\ x_i\in N,i=1,...,p_1\\ x_i\in R_+,i=\left(p_1+1\right),...,p\\ y\in R_+^q\end{array}\right\}\right\}\tag{27}$$

Dari model (27) dapat dilihat

$$\min_{y}\left\{\beta y:\begin{array}{c}Ax+By\geq\max_{\zeta\in\mathbb{Z}}\left\{\overline{d}+p^T\zeta\right\}\\ Cx\geq b\\ x_i\in N,i=1,...,p_1\\ x_i\in R_+,i=\left(p_1+1\right),...,p\\ y\in R_+^q\end{array}\right\}\tag{28}$$

merupakan *Inner Optimization Problem.* Persamaan (28) dapat dituliskan menjadi

$$\max_{\zeta\in\mathbb{Z}}\min_{y}\left\{\beta y,\quad\begin{array}{c}\begin{bmatrix}B\\0\end{bmatrix}y\geq\begin{bmatrix}\left(\overline{d}+p^T\zeta\right)\\ b\end{bmatrix}-\begin{bmatrix}A\\C\end{bmatrix}x\\ x_i\in N,i=1,...,p_1\\ x_i\in R_+,i=\left(p_1+1\right),...,p\\ y\in R_+^q\end{array}\right\}\tag{29}$$

yang dikenal sebagai *Recourse Program*. Merujuk Billionet [2], untuk menyelesaikan model (29) adalah dengan mentransformasikan submasalah minimisasi menjadi maksimisasi dengan program dual menjadi

$$\max_{\zeta\in\mathbb{Z},\lambda}\left\{\left[\begin{bmatrix}\left(\overline{d}+p^T\zeta\right)\\ b\end{bmatrix}-\begin{bmatrix}A\\C\end{bmatrix}x\right]^T\lambda:\begin{bmatrix}B\\0\end{bmatrix}^T\lambda\leq\beta,\lambda\in U\right\}\tag{30}$$

Substitusikan Persamaan (30) ke dalam Persamaan (27), sehingga model menjadi $P_2\left(\lambda,x\right)$:

$$\min_{x}\left\{\alpha x+\max_{\zeta\in\mathbb{Z},\lambda}\left\{\left[\begin{bmatrix}\left(\overline{d}+p^T\zeta\right)\\ b\end{bmatrix}-\begin{bmatrix}A\\C\end{bmatrix}x\right]^T\lambda:\begin{bmatrix}B\\0\end{bmatrix}^T\lambda\leq\beta,\lambda\in U\right\}\right\}\tag{31}$$

karena variabel pada pembatas *Inner Maximum Problem* adalah bebas dan solusi optimumnya adalah tentu, maka model (31) dibentuk menjadi $P_3\left(\lambda,x\right)$:

$$\min_{x}\left\{\alpha x+\max_{\zeta\in\mathbb{Z},\lambda}\left\{\left[\begin{bmatrix}\left(\overline{d}+p^T\zeta\right)\\ b\end{bmatrix}-\begin{bmatrix}A\\C\end{bmatrix}x\right]^T\lambda\right\}\right\}\tag{32}$$

Sehingga model awal Persamaan (26) dapat diubah menjadi sebuah masalah minimisasi tunggal yaitu *full master problem* $P_4\left(x,m\right)$:

$$\min_{x}\left\{\alpha x+m:\left[\begin{bmatrix}\left(\overline{d}+p^T\zeta\right)\\ b\end{bmatrix}-\begin{bmatrix}A\\C\end{bmatrix}x\right]^T\lambda\leq m,\lambda\in U,\zeta\in\mathbb{Z}\right\}\tag{33}$$

dan *relaxed master problem* $M(x,m)$:

$$\min_x \left\{ \alpha x + m : \left[\left[\begin{array}{c} (\overline{d} + p^T\zeta) \\ b \end{array} \right] - \left[\begin{array}{c} A \\ C \end{array} \right] x \right]^T \lambda \leq m, \lambda \in \mathbb{B}, \zeta \in \mathbb{Z} \right\} \tag{34}$$

dimana $\mathbb{B}$ adalah subset dari U, dan submasalah $S(\lambda|x)$:

$$\max_\lambda \left\{ \left[\left[\begin{array}{c} (\overline{d} + p^T\zeta) \\ b \end{array} \right] - \left[\begin{array}{c} A \\ C \end{array} \right] x \right]^T \lambda : \left[\begin{array}{c} B \\ 0 \end{array} \right]^T \lambda \leq \beta, \lambda \geq 0, \zeta \in \mathbb{Z} \right\} \tag{35}$$

karena model (35) dipengaruhi ketidaktentuan pada fungsi objektif, agar memenuhi asumsi fungsi objektif adalah tentu maka dibentuk menjadi

$$\max_f \left\{ f : \left[\left[\begin{array}{c} (\overline{d} + p^T\zeta) \\ b \end{array} \right] - \left[\begin{array}{c} A \\ C \end{array} \right] x \right]^T \lambda \geq f, \left[\begin{array}{c} B \\ 0 \end{array} \right]^T \lambda \leq \beta, \quad \begin{array}{c} f \geq 0 \\ \lambda \geq 0 \\ \forall \zeta \in \mathbb{Z}_i \end{array} \right\} \tag{36}$$

Selanjutnya, mengasumsikan bahwa parameter ketidaktentuan dalam himpunan *ellipsoidal uncertainty*. Didefinisikan himpunan *ellipsoidal uncertainty* berikut:

$$\mathbb{Z} = \{\zeta : \|\zeta\|_2 \leq 1\} \Leftrightarrow \mathbb{Z} = \left\{ \zeta : \sqrt{\sum_i \zeta_i^2} \leq 1 \right\} \tag{37}$$

kemudian misalkan himpunan *uncertainty* $\mathbb{Z}$ dibagi menjadi

$$\begin{aligned} \mathbb{Z}_1 &= \left\{ \zeta : \sqrt{\sum_i \zeta_i^2} = 0 \right\} \\ \mathbb{Z}_2 &= \left\{ \zeta : 0 < \sqrt{\sum_i \zeta_i^2} \leq 1 \right\} \end{aligned} \tag{38}$$

Sehingga untuk $i = 1$ dan $i = 2$ dari (34) yang baru masing-masing menjadi

$$\min_x \left\{ \alpha x + m : \left[\left[\begin{array}{c} (\overline{d}) \\ b \end{array} \right] - \left[\begin{array}{c} A \\ C \end{array} \right] x \right]^T \lambda \leq m, \lambda \in \mathbb{B} \right\} \tag{39}$$

dan

$$\min_x \left\{ \alpha x + m : \left[\left[\begin{array}{c} (\overline{d} + p) \\ b \end{array} \right] - \left[\begin{array}{c} A \\ C \end{array} \right] x \right]^T \lambda \leq m, \lambda \in \mathbb{B} \right\} \tag{40}$$

Sementara untuk $i = 1$ dan $i = 2$ dari (36) yang baru masing-masing menjadi

$$\max_f \left\{ f : \left[\left[\begin{array}{c} (\overline{d}) \\ b \end{array} \right] - \left[\begin{array}{c} A \\ C \end{array} \right] x \right]^T \lambda \geq f, \left[\begin{array}{c} B \\ 0 \end{array} \right]^T \lambda \leq \beta, \begin{array}{c} f \geq 0 \\ \lambda \geq 0 \end{array} \right\} \tag{41}$$

dan

$$\max_f \left\{ f : \left[\left[\begin{array}{c} (\overline{d} + p) \\ b \end{array} \right] - \left[\begin{array}{c} A \\ C \end{array} \right] x \right]^T \lambda \geq f, \left[\begin{array}{c} B \\ 0 \end{array} \right]^T \lambda \leq \beta, \begin{array}{c} f \geq 0 \\ \lambda \geq 0 \end{array} \right\} \tag{42}$$

3.2. **Simulasi Numerik.** Berikut adalah data sekunder yang digunakan pada penelitian ini. Pohon sengon pada satu hektar lahan, dapat menghasilkan 400 pohon untuk masa panen 4-6 tahun, diaman volume per pohon 1.6 m3, dengan harga jual kayu Rp800.000/m3. Sehingga data pada hutan 1, hutan 2, dan hutan 3 pada akhir masa panen 6 tahun (1 periode) dengan asumsi luas hutan 1, hutan 2, dan hutan 3 masing-masing 5 ha, 6 ha, dan 4 ha, dapat dilihat pada Tabel 3.1 berikut:

Diasumsikan biaya penebangan oleh Tim 1 dan Tim 2 masing-masing Rp60.000/m^3 dan Rp50.000/m^3. Sebagai gambaran dari masalah ini perhatikan Gambar 3.1.

Dengan mengguanakan Software Maple untuk pencarian solusi optimal, hasil simulasi numerik untuk model nominal dan robust untuk masalah MILP Dua Tahap dengan

Tabel 1. Data Luas Lahan, Volume Hutan, dan Persediaan

	Luas Lahan	Volume (pohon/ha)	volume (m^3/ha)	Persediaan (pohon)	Persediaan (m^3)
Hutan 1	5	400	640	2000	3200
Hutan 2	6	400	640	2400	3840
Hutan 3	4	400	640	1600	2560

Tabel 2. Data Biaya Penebangan Hutan (dalam rupiah)

	Hutan 1	Hutan 2	Hutan 3
Tim 1	192.000.000	230.400.000	153.600.000
Tim 2	160.000.000	192.000.000	128.000.000

Tabel 3. Data Biaya Transportasi dan Permintaan Kayu (rupiah per m^3)

	Industri 1	Industri 2
Hutan 1	100.000	95.000
Hutan 2	85.000	100.000
Hutan 3	80.000	90.000
Permintaan	3000	2600

Gambar 2. Diagram Perhutanan berbasis CDM berdasarkan Data

Variabel Recourse dapat dilihat pada Tabel 3.4. Dimana terlihat bahwa dengan menggunakan Pendekatan Optimisasi Robust biaya total menjadi lebih besar. Hal ini dikarenakan pada Model Pendekatan Optimisasi Robust ada penambahan penganggu pada parameter ketidaktentuan, yaitu permintaan kayu. Sehingga hasil dari Pendekatan Optimisasi Robust ini sudah Robust terhadap ketidaktentaun pada permintaan kayu dengan asumsi toleransi penganggu sebesar 10%.

Tabel 4. Solusi Optimal dari Model Robust Optimisasi Perhutanan

	Keterangan	Solusi Model Perhutanan	Solusi Model *Robust* Optimisasi Perhutanan	
			$i = 1$	$i = 2$
x_1	Tim 1 menebang Hutan 1	0	0	0
x_2	Tim 2 menebang Hutan 1	0	0	0
x_3	Tim 1 menebang Hutan 2	0	0	0
x_4	Tim 2 menebang Hutan 2	1	1	1
x_5	Tim 1 menebang Hutan 3	1	1	1
x_6	Tim 2 menebang Hutan 3	0	0	0
y_1	Hutan 1 mengirim ke Industri 1	0	0	0
y_2	Hutan 1 mengirim ke Industri 2	0	0	0
y_3	Hutan 2 mengirim ke Industri 1	3000	3000	3300
y_4	Hutan 2 mengirim ke Industri 2	40	40	300
y_5	Hutan 3 mengirim ke Industri 1	0	0	0
y_6	Hutan 3 mengirim ke Industri 2	2560	2560	2560
Z	Biaya Total	Rp835.400.000	Rp835.400.000	Rp886.900.000

3.3. **Analisis Hasil.** Hasil perhitungan numerik model Robust Optimisasi pada Perhutanan adalah sebagai berikut.

Keputusan Strategi Hasil perhitungan untuk masalah nominal dan untuk mengatasi ketidaktentuan pada permintaan kayu menghasilkan solusi yang sama, yaitu Tim 2 menebang pada Hutan 2, dan Tim 1 menebang pada Hutan 3.

Keputusan Operasional Hasil dari masalah nominal didapatkan solusi optimal dimana Hutan 2 mengirimkan kayu ke Industri 1 sebanyak 3000 m^3, Hutan 2 mengirimkan kayu ke Industri 2 sebanyak 40 m^3, dan Hutan 3 mengirimkan kayu sebanyak 2560 m^3. Sedangkan untuk mengatasi ketidaktentuan pada permintaan, maka penebangan hutan dari Tim 1 dan Tim 2 masing-masing pada Hutan 3 dan Hutan 2, banyak kayu yang dikirim dari Hutan 2 ke Industri 1 sebanyak 3300 m^3. Banyak kayu yang dikirim dari Hutan 2 ke Industri 2 sebanyak 300 m^3. Banyak kayu yang dikirim dari Hutan 3 ke Industri 2 sebanyak 2560 m^3.

Biaya Optimum Total biaya minimum penebangan pohon untuk masalah nominal sebesar Rp835.400.000, sedangkan dengan mengatasi ketidaktentuan pada permintaan total biaya minimum penebangan pohon sebesar Rp886.900.000,.

4. Simpulan

Model masalah *Mixed Integer Linear Programming* (MILP) dapat dibentuk dalam model optimisasi *robust* MILP dua tahap dengan variabel *recourse*. Model MILP dibentuk menjadi masalah *robust counterpart*, kemudian didefinisikan *recourse program*. Dalam penyelesaian model optimisasi *robust* MILP dua tahap dengan variabel *recourse* adalah dengan menggunakan Metode *Benders Decomposition*. Dalam penelitian ini diberikan contoh kasus, yaitu optimisasi Perhutanan pada model optimisasi masalah perencanaan taktis perhutanan yang dapat diperoleh solusi optimal *robust*.

Ucapan Terimakasih

Penelitian ini sebagian didanai oleh Penelitian Dasar Kementerian Riset Teknologi dan Pendidikan Tinggi untuk Tahun 2020 No Kontrak 1827/UN6.3.1/LT/2020.

Daftar Pustaka

[1] V. Gabrel, M. Lacroix, C. Murat, N. Remli (2011) *Robust location transportation problems under uncertain demands Discrete Applied Mathematics 164* 100-111.

[2] Alain Billionnet, Marie-Christine Costa, Pierre-Louis Poirion (2014) *2-stage robust MILP with continuous recourse variables Discrete Applied Mathematics 170* 21-32.

[3] Dantzig, George B. (1955) *Linear Programming Under Uncertainty The Rand Corporation, Santa Monica* Cal. Vol. 1

[4] Soyster, A.L. (1973) *Technical Note - Convex Programming with Set-Inclusive Constraints and Applications to Inexact Linear Operations Research 21 (5)* 1154-1157.

[5] A. Ben-Tal, A. Goryashko, E. Guslitzer, A. Nemirovski (2004) *Adjustable robust solutions of uncertain linear programs Math. Program. Ser. A 99* 351-376.

[6] Aurelie Thiele, Tara Terry, Marina Epelman (2010) *Robust Linear Optimization With Recourse*

[7] Gorissen, Ihsan Yanikoglu, Dick den Hertog, Bram L. (2015) *A practical guide to robust optimization Omega Journal Vol. 53* 124-137.

[8] A. Ben-Tal, L. El Ghaoui, A. Nemirovski (2009) *Robust Optimization Princeton Series in Applied Mathematics. Princeton University Press*

[9] Hertog, D. d., Ben-Tal, A. & Brekelmans, R. (2015) *Practical Robust Optimization*

[10] Diah Chaerani, C Roos (2013) *Handling Optimization under Uncertainty Problem Using Robust Jurnal Teknik Industri Vol. 15* 111-118

[11] A. Ben-Tal, A. Nemirovski (2002) *Robust Optimization-Methodology and Aplications Mathematical Programming 92(3)* pp. 453-480.

[12] J. Bisschop (1999) *AIMMS Optimization Modeling. Netherlands: AIMMS B.V*