

II. TINJAUAN PUSTAKA

2.1. Persamaan Diferensial Parsial

Suatu persamaan yang meliputi turunan fungsi dari satu atau lebih variabel terikat terhadap satu atau lebih variabel bebas disebut persamaan diferensial. Jika turunan fungsi itu hanya tergantung pada satu variabel bebas maka disebut persamaan diferensial biasa (PDB) dan bila tergantung pada lebih dari satu variabel bebas disebut persamaan diferensial parsial (PDP). Pada PDP, variabel bebas dapat berupa waktu dan satu atau lebih koordinat ruang. (Soedradjat, 2003).

2.2. Persamaan Diferensial Parsial dengan Metode Beda Hingga

PDP adalah persamaan diferensial dengan dua variabel bebas / penentu atau lebih.

Telah dibahas Definisi Turunan pertama:

- **Beda Maju:**

$$\frac{dy}{dx} = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}$$

- **Beda Mundur :**

$$\frac{dy}{dx} = \lim_{h \rightarrow 0} \frac{f(x) - f(x-h)}{h}$$

- **Beda Tengah :**

$$\frac{dy}{dx} = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x-h)}{2h}$$

dan

$$\frac{d^2y}{dx^2} = \lim_{h \rightarrow 0} \frac{f(x+h) + f(x-h) - 2f(x)}{h^2}$$

Berdasarkan definisi tersebut, maka dapat diketahui definisi dari turunan parsial sebagai berikut:

$$\begin{aligned}
 \text{Beda Maju} & : \frac{\partial f}{\partial x} = \lim_{h \rightarrow 0} \frac{f(x+h, y) - f(x, y)}{h} & \text{dan} & \frac{\partial f}{\partial y} = \lim_{h \rightarrow 0} \frac{f(x, y+h) - f(x, y)}{h} \\
 \text{Beda Mundur} & : \frac{\partial f}{\partial x} = \lim_{h \rightarrow 0} \frac{f(x, y) - f(x-h, y)}{h} & \text{dan} & \frac{\partial f}{\partial y} = \lim_{h \rightarrow 0} \frac{f(x, y) - f(x, y-h)}{h} \\
 \text{Beda Tengah} & : \frac{\partial f}{\partial x} = \lim_{h \rightarrow 0} \frac{f(x+h, y) - f(x-h, y)}{2h} & \text{dan} & \frac{\partial f}{\partial y} = \lim_{h \rightarrow 0} \frac{f(x, y+h) - f(x, y-h)}{2h}
 \end{aligned}$$

Dan definisi turunan Parsial Tingkat Dua.

$$\begin{aligned}
 \frac{\partial^2 f}{\partial x^2} & = \lim_{h \rightarrow 0} \frac{f(x+h, y) + f(x-h, y) - 2f(x, y)}{h^2} & \text{Dan} \\
 \frac{\partial^2 f}{\partial y^2} & = \lim_{h \rightarrow 0} \frac{f(x, y+h) + f(x, y-h) - 2f(x, y)}{h^2}
 \end{aligned}$$

(Anonymous, 2014)

2.3 Metode Beda Hingga

Metode beda hingga adalah metode numerik yang umum digunakan untuk menyelesaikan persoalan teknis dan problem matematis dari suatu gejala fisis. Secara umum metode beda hingga adalah metode yang mudah digunakan dalam penyelesaian problem fisis yang mempunyai bentuk geometri yang teratur, seperti interval dalam satu dimensi, domain kotak dalam dua dimensi, dan kubik dalam ruang tiga dimensi (Li, 2010).

Aplikasi penting dari metode beda hingga adalah dalam analisis numerik, khususnya pada persamaan diferensial biasa dan persamaan diferensial parsial. Prinsipnya adalah mengganti turunan yang ada pada persamaan diferensial dengan diskritisasi beda hingga berdasarkan deret Taylor. Secara fisis, deret Taylor dapat

diartikan sebagai besaran tinjauan pada suatu ruang dan waktu (ruang dan waktu tinjauan) dapat dihitung dari besaran itu sendiri pada ruang dan waktu tertentu yang mempunyai perbedaan yang kecil dengan ruang dan waktu tinjauan.

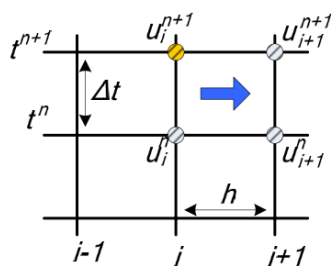
Metode beda hingga bersifat eksplisit, artinya keadaan suatu sistem atau solusi variabel pada suatu saat dapat digunakan untuk menentukan keadaan sistem pada waktu berikutnya. Berbeda dengan metode implisit, yang mana penentuan solusi sistem harus dengan memecahkan sistem pada kedua keadaan, sekarang dan yang akan datang.

Berdasarkan ekspansi Taylor di atas, terdapat tiga skema beda hingga yang biasa digunakan yaitu skema maju, skema mundur, dan skema tengah.

1. Skema maju

$$\frac{\partial u}{\partial x} = \frac{u(x_i + h) - u(x_i)}{h}$$

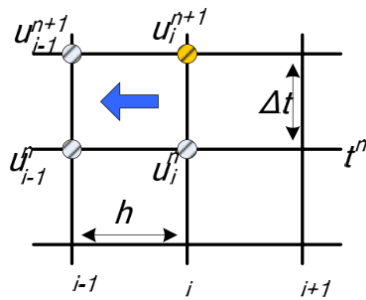
Pada skema maju, informasi pada titik hitung i dihubungkan dengan titik hitung $i+1$ yang berada di depannya.



2. Skema mundur

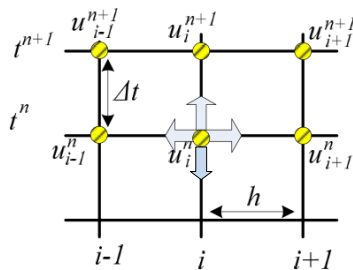
$$\frac{\partial u}{\partial x} = \frac{u(x_i) - u(x_i - h)}{h}$$

Pada skema mundur, informasi pada titik hitung i dihubungkan dengan titik hitung $(i-1)$ yang berada di belakangnya.



3. Skema tengah

$$\frac{\partial u}{\partial x} = \frac{u(x_i + h) - u(x_i - h)}{2h}$$



(Anderson, 1984).

2.4. Pemodelan matematika

Pemodelan matematika merupakan proses dalam menurunkan model matematika dari suatu fenomena berdasarkan asumsi-asumsi yang digunakan. Proses ini merupakan langkah awal yang tak terpisahkan dalam menerapkan matematika untuk mempelajari fenomena-fenomena alam, ekonomi, sosial maupun fenomena-fenomena lainnya. Secara umum dalam menerapkan matematika untuk mempelajari suatu fenomena meliputi 3 langkah, yaitu :

1. Pemodelan matematika suatu fenomena, perumusan masalah.

Langkah ini untuk menterjemahkan data maupun informasi yang diperoleh tentang suatu fenomena dari masalah nyata menjadi model matematika. Data maupun informasi tentang suatu fenomena dapat diperoleh melalui eksperimen di laboratorium, pengamatan di industri ataupun dalam kehidupan sehari-hari. Dalam

model matematika, suatu fenomena dapat dipelajari secara lebih terukur (kuantitatif) dalam bentuk (sistem) persamaan/pertidaksamaan matematika maupun ekspresi matematika. Namun demikian karena asumsi-asumsi yang digunakan dalam prosesnya, model matematika juga mempunyai kelemahan-kelemahan dibandingkan dengan fenomena sebenarnya, yaitu keterbatasan dalam generalisasi interpretasinya.

2. Pencarian solusi/kesimpulan matematika.

Setelah model matematika diperoleh, solusi atas model tersebut dicari dengan menggunakan metode-metode matematika yang sesuai. Ada kalanya belum terdapat metode matematika pencarian solusi yang sesuai dengan permasalahan yang dihadapi. Hal ini sering menjadi motivasi para ahli matematika terapan untuk menciptakan metode matematika baru. Solusi matematika ini sering dinyatakan dalam fungsi-fungsi matematika, angka-angka maupun grafik.

3. Interpretasi solusi/kesimpulan matematika pada fenomena yang dipelajari.

Dalam matematika terapan, solusi yang berupa fungsi, angka-angka maupun grafik tidak berarti banyak apabila solusi tersebut tidak menjelaskan permasalahan awalnya. Oleh karena itu, interpretasi solusi penting untuk mengerti arti dan implikasi solusi tersebut terhadap fenomena awal dari mana masalahnya berasal (*Cahyono, 2013*).

Model adalah representasi penyederhanaan dari sebuah realita yang kompleks (biasanya bertujuan untuk memahami realita tersebut) dan mempunyai *feature* yang sama dengan tiruannya dalam melakukan *task* atau menyelesaikan permasalahan. Model adalah karakteristik umum yang mewakili sekelompok

bentuk yang ada, atau representasi suatu masalah dalam bentuk yang lebih sederhana dan mudah dikerjakan. Dalam matematika, teori model adalah ilmu yang menyajikan konsep-konsep matematis melalui konsep himpunan, atau ilmu tentang model-model yang mendukung suatu sistem matematis. Teori model diawali dengan asumsi keberadaan obyek-obyek matematika (misalnya keberadaan semua bilangan) dan kemudian mencari dan menganalisis keberadaan operasi-operasi, relasi-relasi, atau aksioma-aksioma yang melekat pada masing-masing obyek atau pada obyek-obyek tersebut. Independensi dua hukum matematis yang lebih dikenal dengan nama *axiom of choice*, dan *continuum hypothesis* dari aksioma-aksioma teori himpunan (dibuktikan oleh Paul Cohen dan Kurt Godel) adalah dua hasil terkenal yang diperoleh dari teori model.

Telah dibuktikan bahwa *axiom of choice* dan negasinya konsisten dengan aksioma-aksioma Zermelo- Fraenkel dalam teori himpunan dan hasil yang sama juga dipenuhi oleh *continuum hypothesis*. Model matematika yang diperoleh dari suatu masalah matematika yang diberikan, selanjutnya diselesaikan dengan aturan-aturan yang ada. Penyelesaian yang diperoleh, perlu diuji untuk mengetahui apakah penyelesaian tersebut valid atau tidak. Hasil yang valid akan menjawab secara tepat model matematikanya dan disebut solusi matematika. Jika penyelesaian tidak valid atau tidak memenuhi model matematika maka solusi masalah belum ditemukan, dan perlu dilakukan pemecahan ulang atas model matematikanya. (*Frederich H. Bell 1978*).

Pemodelan matematika adalah penyusunan suatu deskripsi dari beberapa perilaku dunia nyata (fenomena-fenomena alam) ke dalam bagian-bagian matematika yang

disebut dunia matematika (*mathematical world*). Pemodelan matematika juga merupakan representasi dari objek, proses, atau hal lain yang diharapkan dapat diketahui polanya sehingga dapat dianalisis.

(Dym and Ivey, 1980)

2.5 Angin

Angin adalah udara yang bergerak yang diakibatkan oleh rotasi bumi, dan juga karena adanya perbedaan tekanan udara disekitarnya. Angin bergerak dari tempat bertekanan udara yang tinggi ke tempat yang bertekanan udara rendah. Apabila dipanaskan, udara memuai. Udara yang telah memuai menjadi lebih ringan sehingga naik. Apabila hal ini terjadi, tekanan udara turun karena udaranya berkurang.

Beberapa faktor terjadinya angin diantaranya makin besar gradien barometrisnya, makin cepat tiupan angin, kecepatan angin di dekat khatulistiwa lebih cepat dari yang jauh dari garis khatulistiwa, semakin tinggi tempat maka semakin kencang pula angin yang bertiup, hal ini disebabkan oleh pengaruh gaya gesekan yang menghambat laju udara. Di permukaan bumi, gunung, pohon, dan topografi yang tidak rata lainnya memberikan gaya gesekan yang besar. Semakin tinggi suatu tempat, gaya gesekan ini semakin kecil serta di siang hari angin bergerak lebih cepat daripada di malam hari.

(Daryanto, 2000).

2.6. LINDO (*Linear Ineraktive Discrete Optimizer*)

Ada banyak *software* yang digunakan untuk menyelesaikan masalah pemrograman linear seperti TORA, LINGO, EXCEL dan banyak lagi yang lainnya. Adapun salah satu *software* yang sangat mudah digunakan untuk masalah pemrograman linear adalah dengan menggunakan Lindo.

Lindo (*Linear Ineraktive Discrete Optimizer*) adalah *software* yang dapat digunakan untuk mencari penyelesaian dari masalah pemrograman linear. Dengan menggunakan *software* ini memungkinkan perhitungan masalah pemrograman linear dengan n variabel. Prinsip kerja utama Lindo adalah memasukkan data, menyelesaikan, serta menaksirkan kebenaran dan kelayakan data berdasarkan penyelesaiannya. Menurut Linus Scharge (1991), perhitungan yang digunakan pada Lindo pada dasarnya menggunakan metode simpleks. Sedangkan untuk menyelesaikan masalah pemrograman linear *integer* nol-satu *software* Lindo menggunakan Metode *Branch and Bound* (metode Cabang dan Batas) menurut Mark Wiley (2010). Untuk menentukan nilai optimal dengan menggunakan Lindo diperlukan beberapa tahapan yaitu:

1. Menentukan model matematika berdasarkan data real
2. Menentukan formulasi program untuk Lindo
3. Membaca hasil *report* yang dihasilkan oleh Lindo.

Perintah yang biasa digunakan untuk menjalankan program Lindo adalah:

1.	MAX	digunakan untuk memulai data dalam masalah maksimasi;
2.	MIN	digunakan untuk memulai data dalam masalah minimasi;
3.	END	digunakan untuk mengakhiri data;

4.	GO	digunakan untuk pemecahan dan penyelesaian masalah;
5.	LOOK	digunakan untuk mencetak bagian yang dipilih dari data yang ada;
6.	GIN	digunakan untuk variabel keputusan agar bernilai bulat;
7.	INTE	digunakan untuk menentukan solusi dari masalah biner;
8.	INT	sama dengan INTE;
9.	SUB	digunakan untuk membatasi nilai maksimumnya;
10.	SLB	digunakan untuk membatasi nilai minimumnya;
11.	FREE	digunakan agar solusinya berupa bilangan real.

Kegunaan utama dari program Lindo adalah untuk mencari penyelesaian dari masalah linier dengan cepat dengan memasukan data yang berupa rumusan dalam bentuk linier. Lindo memberikan banyak manfaat dan kemudahan dalam memecahkan masalah optimasi dan minimasi. Berikut ini cara memulai menggunakan program Lindo adalah dengan membuka file Lindo kemudian klik dua kali pada Lindow32, tunggu sampai muncul dialog lalu klik OK, Lindo siap dioperasikan.

Pada layar akan muncul *untitled* baru yang siap untuk tempat menetikkan formasi.

Model Lindo minimal memiliki tiga syarat:

1. Memerlukan fungsi objektif;
2. Variabel;
3. Batasan (fungsi kendala).

Untuk syarat pertama fungsi objektif, bisa dikatakan tujuan. Tujuan disini memiliki dua jenis tujuan yaitu maksimasi (*MAX*) dan minimasi (*MIN*). Kata pertama untuk mengawali pengetikan formula pada Lindo adalah *MAX* atau *MIN*. Formula yang diketikkan ke dalam *untitled* (papan *editor* pada Lindo) setelah *MAX* atau *MIN* disebut fungsi tujuan. Secara umum dapat dituliskan sebagai berikut.

Fungsi tujuan model matematika

$$\text{Min/Maks } Z = C_1X_1 + C_2X_2 + \dots + C_nX_n$$

Diketikkan ke dalam *untitled* menjadi

$$\text{MIN } C_1X_1 + C_2X_2 + \dots + C_nX_n$$

atau

$$\text{MAX } C_1X_1 + C_2X_2 + \dots + C_nX_n$$

Untuk syarat kedua adalah variabel. Variabel ini sangat penting, Lindo tidak dapat dijalankan tanpa memasukkan variabel dalam formula.

Untuk syarat ketiga setelah fungsi objektif dan variabel selanjutnya adalah batasan. Dalam kenyataannya variabel tersebut pasti memiliki batasan, batasan itu misalnya keterbatasan bahan, waktu, jumlah pekerja, biaya operasional. Setelah fungsi objektif diinput selanjutnya diinput *Subject to* atau *ST* untuk mengawali penginputan batasan dan pada baris berikutnya baru diinput batasan yang ada diakhir batasan di akhiri dengan kata *END*. Secara umum dapat dituliskan sebagai berikut.

$$a_{11}X_1 + a_{12}X_2 + \dots + C_{1n}X_n \leq b_1$$

$$a_{11}X_1 + a_{22}X_2 + \dots + C_{2n}X_n \leq b_2$$

$$a_{m1}X_1 + a_{m2}X_2 + \dots + C_{mn}X_n \leq b_m$$

$$X_1, X_2, \dots, X_n \geq 0$$

untuk pengetikkan fungsi kendala ke dalam *untitled* adalah sebagai berikut.

SUBJECT TO

$$a_{11}X_1 + a_{12}X_2 + \dots + C_{1n}X_n \leq b_1$$

$$a_{11}X_1 + a_{22}X_2 + \dots + C_{2n}X_n \leq b_2$$

$$a_{m1}X_1 + a_{m2}X_2 + \dots + C_{mn}X_n \leq b_m$$

$$X_1 \geq 0$$

$$X_2 \geq 0$$

$$X_n \geq 0$$

END

Setelah formula diinput siap dicari solusinya dengan memilih perintah *solve* atau melanjutkan tombol *solve* pada *toolbar*. Lindo akan mengkompil (mengoreksi kesalahan) pada formula terlebih dahulu. Jika terjadi kesalahan dalam penginputan (tidak dapat dibaca oleh komputer) akan muncul kotak dialog dan kursor akan menunjukkan pada baris yang salah.

Menu Solve

Menu *solve* digunakan untuk menampilkan hasil secara lengkap dengan beberapa pilihan berikut:

1. *Solve-Solve*, digunakan untuk menampilkan hasil optimasi dari data pada papan *editor* dan secara lengkap. Pada tampilan hasil mencakup nilai variabel keputusan serta nilai *dual price*-nya. Pada nilai peubah keputusan ditampilkan pula nilai peubah keputusan yang nol. Perbedaannya dengan *Report Solusion* adalah pada *Report Solusion* kadang-kadang jawabannya tidak optimal iterasinya, sehingga pada *Solve-Solve* jawaban yang ditampilkan bernilai optimal. *Report Solution* tidak menampilkan nilai *Dual Price* serta ada pilihan apakah perlu ditampilkan nilai peubah keputusan yang nol.
2. *Solve-Compile Model*, digunakan untuk mengecek apakah struktur penyusunan data pada papan editor data sudah benar. Jika penulisannya tidak benar, maka akan ditampilkan pada baris ke-berapa kesalahan tersebut terdapat. Jika tidak ada kesalahan, maka proses dapat dilanjutkan untuk mencari jawaban yang optimal.
3. *Solve Pivot*, digunakan untuk menampilkan nilai slack.
4. *Solve Debug*, digunakan untuk mempersempit permasalahan serta mencari pada bagian mana yang mengakibatkan solusi tidak optimal, selanjutnya ada pertanyaan untuk menentukan tingkat kesensitifitasan solusi.

Jika tidak terjadi kesalahan akan muncul status *Lindo*. Status ini berguna untuk memonitor proses solusi. Selanjutnya tekan *close* dan pada *Lindo* akan muncul tampilan baru yang disebut *report windows*.

Untuk tampilan pada *report* diatur sesuai dengan kebutuhan. Pengaturan *report* dilakukan dengan memilih *Report* pada *toolbar* Lindo.

Dalam menu *report* terdapat beberapa pilihan sebagai berikut:

1. *Report Solution*, digunakan untuk mendapatkan solusi optimal dari permasalahan program linier yang tersaji pada papan editor data.
2. *Report Range*, digunakan untuk menayangkan hasil penyelesaian analisis sensitivitas. Pada analisis sensitivitas yang ditayangkan mencakup aspek *Allowable Increase* dan *Allowable Decrease*.
3. *Report Parametrics*, digunakan untuk mengubah dan menampilkan hasil hanya pada baris kendala tertentu saja.
4. *Report Statistics*, digunakan untuk mendapatkan laporan kecil pada papan *editor report*.
5. *Report Peruse*, digunakan untuk menampilkan sebagian dari model atau jawaban.
6. *Report Picture*, digunakan untuk menampilkan (display) model dalam bentuk matriks.
7. *Report Basis Picture*, digunakan untuk menampilkan text format dari nilai basis, dan disajikan sesuai urutan baris dan kolom.
8. *Report Table*, digunakan untuk menampilkan tabel simplek dari model yang ada.
9. *Report Formulation*, digunakan untuk menampilkan model pada papan *editor data* ke papan *editor report*.
10. *Report Show Coloum*, digunakan untuk menampilkan koefisien peubah.

Untuk menyimpan *file*, arahkan kursor pada papan *editor* yang diaktifkan. Menu menyimpan *file* ada dua macam yakni *File Save*, dan *File Save As*. (**Linus Schrange, 1991**).