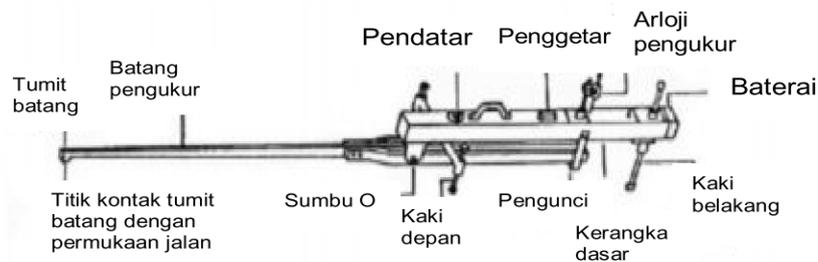


BAB III LANDASAN TEORI

3.1 *Benkelman Beam*

Menurut Pedoman Perencanaan Tebal Lapis Tambahan Perkerasan Lentur dengan Metode Lendutan Pd. T-05-2005-B, *Benkelman Beam* merupakan alat untuk mengukur lendutan balik dan lendutan langsung perkerasan yang menggambarkan kekuatan struktur perkerasan jalan. Lendutan balik (*rebound deflection*) adalah besar lendutan balik vertikal suatu permukaan perkerasan akibat beban yang berpindah dan lendutan langsung (*directly deflection*) adalah besar lendutan vertikal suatu permukaan perkerasan akibat beban langsung. Tebal lapis tambah (*Overlay*) merupakan lapis perkerasan tambahan yang dipasang diatas konstruksi perkerasan yang ada dengan tujuan untuk meningkatkan kekuatan struktur perkerasan yang ada agar dapat melayani lalu lintas yang direncanakan selama kurun waktu yang akan datang.

Menurut Metode Pengujian Lendutan Perkerasan Lentur dengan Alat *Benkelman Beam* SNI 03-2416-1991, Metode ini dimaksudkan sebagai pegangan dalam pengujian perkerasan jalan dengan alat *Benkelman Beam* (BB) yaitu mengukur gerakan vertikal pada permukaan lapis jalan dengan cara mengatur pemberian beban roda yang diakibatkan oleh beban tertentu dengan tujuan untuk memperoleh data lapangan yang akan bermanfaat bagi penilaian struktur peramalan performance perkerasan dan perencanaan *Overlay*. Bagian-bagian dari alat *Benkelman Beam* dapat dilihat pada Gambar 3.1 berikut.



Gambar 3.1 Alat *Benkelman Beam*

(Sumber: Badan Standarisasi Nasional, 2011)

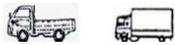
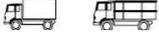
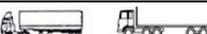
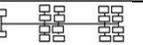
3.2 Perencanaan Tebal Lapis Tambah Berdasarkan Manual Perkerasan Jalan Nomor 04/SE/Db/2017

3.2.1 Analisis Lalu Lintas

1. Jenis Kendaraan

Perhitungan jumlah lalu lintas harus menggunakan formulir baku yang terdiri dari 8 golongan kendaraan yang dapat dilihat pada tabel 3.1:

Tabel 3.1 Konfigurasi Beban Sumbu Kendaraan

Golongan	Kelompok jenis kendaraan	Jenis Kendaraan	Konfigurasi Sumbu	Kode
1	Sepeda motor, kendaraan roda 3			
2	Sedan, jeep, station wagon			1.1
3	Angkatan penumpang sedang			1.1
4	Pick up, micro truk dan mobil hantaran			1.1
5a	Bus kecil			1.1
5b	Bus besar			1.2
6a	Truk ringan 2 sumbu			1.1
6b	Truk sedang 2 sumbu			1.2
7a	Truk 3 sumbu			1.2.2
7b	Truk gandeng			1.2.2 – 2.2
7c	Truk semitrailer			1.2.2.2.2
8	Kendaraan tidak bermotor			

Sumber: Direktorat Jenderal Bina Marga, 2017

Untuk desain tebal perkerasan hanya diperlukan data lalu lintas dari golongan 2 sampai 7. Untuk menentukan kapasitas suatu jalan dengan cara simplified juga diperlukan data hanya lalu lintas dari golongan 2 sampai golongan 7.

2. Faktor Pertumbuhan Lalu Lintas

Faktor pertumbuhan lalu lintas berdasarkan data–data pertumbuhan series (*historical growth data*) atau formulasi korelasi dengan faktor pertumbuhan lain yang berlaku. Jika tidak tersedia data maka Tabel 3.2. dapat digunakan (2015 – 2035).

Tabel 3.2 Faktor Laju Pertumbuhan Lalu Lintas (*i*) (%)

	Jawa	Sumatera	Kalimantan	Rata-rata Indonesia
Arteri dan perkotaan	4,80	4,83	5,14	4,75
Kolektor rural	3,50	3,50	3,50	3,50
Jalan desa	1,00	1,00	1,00	1,00

Sumber: Direktorat Jenderal Bina Marga, 2017

Pertumbuhan lalu lintas selama umur rencana dihitung dengan faktor pertumbuhan kumulatif (*Cumulative Growth Factor*):

$$R = \frac{(1+0,01 i)^{UR}-1}{0,01 i} \quad (3.1)$$

Keterangan:

R = faktor pengali pertumbuhan lalu lintas kumulatif

i = laju pertumbuhan lalu lintas tahunan (%)

UR = umur rencana (tahun)

3. Lalu Lintas Pada Lajur Rencana

Lajur rencana adalah salah satu lajur lalu lintas dari suatu ruas jalan yang menampung lalu lintas kendaraan niaga (truk dan bus) paling besar. Beban lalu

lintas pada lajur rencana dinyatakan dalam kumulatif beban gandar standar (ESA) dengan memperhitungkan faktor distribusi arah (DD) dan faktor distribusi lajur kendaraan niaga (DL).

Untuk jalan dua arah, faktor distribusi arah (DD) umumnya diambil 0,50 kecuali pada lokasi - lokasi yang jumlah kendaraan niaga cenderung lebih tinggi pada satu arah tertentu.

Faktor distribusi lajur digunakan untuk menyesuaikan beban kumulatif (ESA) pada jalan dengan dua lajur atau lebih dalam satu arah. Pada jalan yang demikian, walaupun sebagian besar kendaraan niaga akan menggunakan lajur luar, sebagian lainnya akan menggunakan lajur-lajur dalam. Faktor distribusi jalan yang ditunjukkan pada Tabel 3.3.

Tabel 3.3 Faktor Distribusi Lajur (DL)

Jumlah Lajur Setiap Arah	Kendaraan Niaga pada lajur desain (% terhadap populasi kendaraan niaga)
1	100
2	80
3	60
4	50

Sumber: Direktorat Jenderal Bina Marga, 2017

4. Faktor Ekuivalen Beban / *Vehicle Damage Factor (VDF)*

Dalam desain perkerasan, beban lalu lintas dikonversi ke beban standar (ESA) dengan menggunakan Faktor Ekuivalen Beban (*Vehicle Damage Factor*). Analisis struktur perkerasan dilakukan berdasarkan jumlah kumulatif ESA pada lajur rencana sepanjang umur rencana.

Jika survei beban gandar tidak mungkin dilakukan oleh perencana dan data survei beban gandar sebelumnya tidak tersedia, maka nilai *VDF* pada tabel 3.4 dan tabel 3.5 dapat digunakan untuk menghitung ESA.

Tabel 3.4 Nilai VDF Masing – masing Jenis Kendaraan Niaga

Jenis Kendaraan	Sumatera				Jawa				Kalimantan				Sulawesi				Bali, Nusa Tenggara, Maluku dan Papua			
	Beban aktual		Normal		Beban aktual		Normal		Beban aktual		Normal		Beban aktual		Normal		Beban aktual		Normal	
	VDF4	VDF5	VDF4	VDF5	VDF4	VDF5	VDF4	VDF5												
5B	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0
6A	0,55	0,5	0,55	0,5	0,55	0,5	0,55	0,5	0,55	0,5	0,55	0,5	0,55	0,5	0,55	0,5	0,55	0,5	0,55	0,5
6B	4,5	7,4	3,4	4,6	5,3	9,2	4,0	5,1	4,8	8,5	3,4	4,7	4,9	9,0	2,9	4,0	3,0	4,0	2,5	3,0
7A1	10,1	18,4	5,4	7,4	8,2	14,4	4,7	6,4	9,9	18,3	4,1	5,3	7,2	11,4	4,9	6,7	-	-	-	-
7A2	10,5	20,0	4,3	5,6	10,2	19,0	4,3	5,6	9,6	17,7	4,2	5,4	9,4	19,1	3,8	4,8	4,9	9,7	3,9	6,0
7B1	-	-	-	-	11,8	18,2	9,4	13,0	-	-	-	-	-	-	-	-	-	-	-	-
7B2	-	-	-	-	13,7	21,8	12,6	17,8	-	-	-	-	-	-	-	-	-	-	-	-
7C1	15,9	29,5	7,0	9,6	11,0	19,8	7,4	9,7	11,7	20,4	7,0	10,2	13,2	25,5	6,5	8,8	14,0	11,99	10,2	8,0
7C2A	19,8	39,0	6,1	8,1	17,1	33,0	7,6	10,2	8,2	14,7	4,0	5,2	20,2	42,0	6,6	8,5	-	-	-	-
7C2B	20,7	42,8	6,1	8,0	13,4	24,2	6,5	8,5	-	-	-	-	17,0	28,8	9,3	13,5	-	-	-	-
7C3	24,5	51,7	6,4	8,0	18,1	34,4	6,1	7,7	13,5	22,9	9,8	15,0	28,7	59,6	6,9	8,8	-	-	-	-

Sumber: Direktorat Jenderal Bina Marga, 2017

Tabel 3.5 Nilai VDF Masing – masing Jenis Kendaraan Niaga

Jenis Kendaraan		Uraian	Konfigurasi Sumbu	Muatan yang diangkut	Kelompok Sumbu	Distribusi tipikal (%)		Faktor Ekuivalen Beban (VDF) (ESA/kendaraan)	
Klasifikasi Lama	Alternatif					Semua kendaraan bermotor	Semua kendaraan bermotor kecuali sepeda motor	VDF4 Pangkat 4	VDF5 Pangkat 5
1	1	Sepeda motor	1.1		2	30,4			
2, 3, 4	2, 3, 4	Sedan / Angkot / Pick up / Station wagon	1.1		2	51,7	74,3		
5a	5a	Bus kecil	1.2		2	3,5	5,00	0,3	0,2
5b	5b	Bus Besar	1.2		2	0,1	0,20	1,0	1,0
6a.1	6.1	Truk 2 sumbu – cargo ringan	1.1	Muatan umum	2			0,3	0,2
6a.2	6.2	Truk 2 sumbu – ringan	1.2	Tanah, pasir, besi, semen	2	4,6	6,60	0,8	0,8
6b1.1	7.1	Truk 2 sumbu – cargo sedang	1.2	Muatan umum	2			0,7	0,7
6b1.2	7.2	Truk 2 sumbu - sedang	1.2	Tanah, pasir, besi, semen	2	-	-	1,6	1,7
6b2.1	8.1	Truk 2 sumbu - berat	1.2	Muatan umum	2			0,9	0,8
6b2.2	8.2	Truk 2 sumbu – berat	1.2	Tanah, pasir, besi, semen	2	3,8	5,50	7,3	11,2
7a1	9.1	Truk 3 sumbu – ringan	1.22	Muatan umum	3			7,6	11,2
7a2	9.2	Truk 3 sumbu – sedang	1.22	Tanah, pasir, besi, semen	3	3,9	5,60	28,1	64,4
7a3	9.3	Truk 3 sumbu – berat	1.1.2		3	0,1	0,10	28,9	62,2
7b	10	Truk 2 sumbu dan trailer penarik 2 sumbu	1.2-2.2		4	0,5	0,70	36,9	90,4
7c1	11	Truk 4 sumbu – trailer	1.2-22		4	0,3	0,50	13,6	24,0
7c2.1	12	Truk 5 sumbu – trailer	1.2-22		5			19,0	33,2
7c2.2	13	Truk 5 sumbu – trailer	1.2-222		5	0,7	1,00	30,3	69,7
7c3	14	Truk 6 sumbu - trailer	1.22-222		6	0,3	0,50	41,6	93,7

Sumber: Direktorat Jenderal Bina Marga, 2017

Desain tebal perkerasan didasarkan pada nilai ESA pangkat 4 dan pangkat 5 tergantung pada model kerusakan (*deterioration model*) dan pendekatan desain yang digunakan. Gunakan nilai ESA yang sesuai sebagai input dalam proses perencanaan.

- a. Pangkat 4 digunakan pada desain perkerasan lentur berdasarkan Pedoman Perencanaan Tebal Perkerasan Lentur Pt T-01-2002-B atau metode AASHTO 1993 (pendekatan statistik empirik).
- b. Pangkat 4 digunakan untuk bagan desain pelaburan tipis (seperti Burtu atau Burda), perkerasan tanpa penutup (*Unsealed granular pavement*) dan perencanaan tebal overlay berdasarkan grafik lendutan untuk kriteria alur (*rutting*).
- c. Pangkat 5 digunakan untuk desain perkerasan lentur (kaitannya dengan faktor kelelahan aspal beton dalam desain dengan pendekatan Mekanistik Empiris) termasuk perencanaan tebal overlay berdasarkan grafik lengkung lendutan (*curvature curve*) untuk kriteria retak lelah (*fatigue*).

5. Beban Sumbu Standar Kumulatif

Beban sumbu standar kumulatif atau *Cumulative Equivalent Single Axle Load* (CESAL) merupakan jumlah kumulatif beban sumbu lalu lintas desain pada lajur desain selama umur rencana, yang ditentukan sebagai berikut:

$$ESA_{TH-1} = (\sum LHR_{JK} \times VDF_{JK}) \times 365 \times DD \times DL \times R \quad (3.2)$$

Keterangan:

ESA_{TH-1} : kumulatif lintasan sumbu standar ekivalen (equivalent standard axle) pada tahun pertama.

LHR_{JK} : lintas harian rata – rata tiap jenis kendaraan niaga (satuan kendaraan per hari).

VDF_{JK} : Faktor Ekivalen Beban (Vehicle Damage Factor) tiap jenis kendaraan niaga.

DD : Faktor distribusi arah

DL : Faktor distribusi lajur

- CESAL* : Kumulatif beban sumbu standar ekivalen selama umur rencana.
 R : Faktor pengali pertumbuhan lalu lintas kumulatif

Terdapat tiga prosedur tebal overlay berdasarkan beban lalu lintas.

1. Lalu Lintas lebih kecil atau sama dengan 100.000 ESA4

Retak leleh bukan merupakan kerusakan yang umum terjadi pada jalan dengan lalu lintas ringan dan perkerasan dengan HRS. Berdasarkan pertimbangan itu, desain jalan dengan beban lalu lintas rencana lebih kecil dari 100.000 ESA4 dan perkerasan dengan HRS kinerja *fatigue overlay* tidak diperlukan. Desain tebal overlay cukup dengan pendekatan lendutan maksimum (D_0) sesuai solusi berdasarkan Gambar 3.2.

2. Lalu Lintas lebih besar dari 100.000 ESA4

Pada jalan dengan lalu lintas lebih besar dari 100.000 ESA4 terdapat potensi retak leleh lapisan aspal. Dengan demikian, kriteria deformasi permanen (pendekatan lendutan maksimum D_0) dan kriteria retak leleh (pendekatan lengkung lendutan, $D_0 - D_{200}$) harus diperhitungkan. Gunakan grafik desain Gambar 3.2 dan Gambar 3.4 serta Gambar 3.5

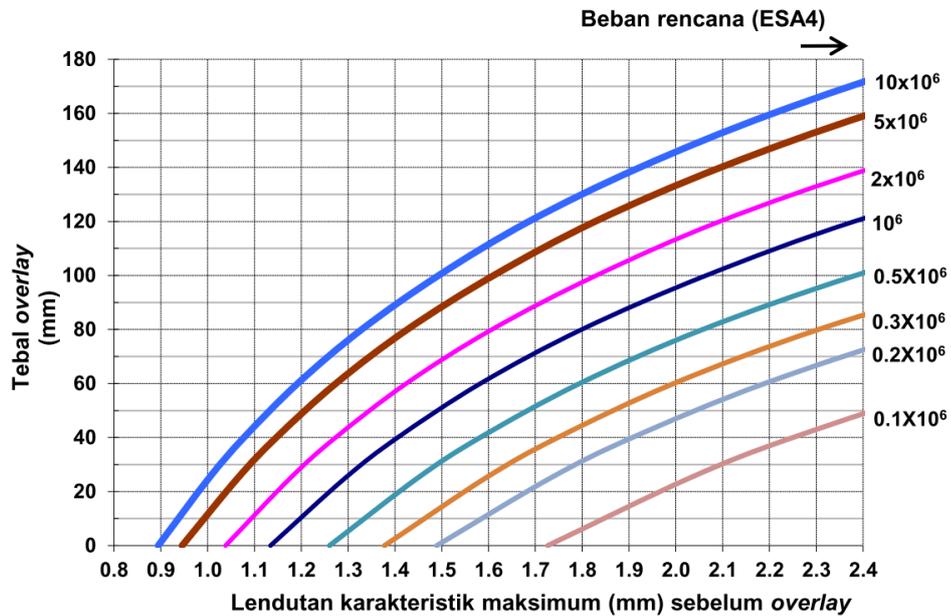
3. Lalu Lintas lebih besar 10×10^6 ESA4 atau 20×10^6 ESA5

Untuk pekerjaan rehabilitasi dengan beban lalu lintas lebih besar daripada 10×10^6 ESA4 atau lebih besar daripada 20×10^6 ESA5 harus digunakan prosedur mekanistik empiris atau metode metode Pt T-01-2002-B atau metode AASHTO 1993.

3.2.2 Tebal Overlay Berdasarkan Lendutan Maksimum

Untuk menentukan tebal overlay berdasarkan lendutan balik maksimum (yang diukur dengan alat *Benkelman Beam*). Hitung dan masukkan nilai lendutan karakteristik dan beban lalu lintas desain (ESA4) pada Gambar 3.2.

Gambar 3.2. digunakan untuk menentukan kebutuhan overlay untuk mengantisipasi deformasi permanen. Desain berdasarkan Gambar 3.2. menghasilkan desain dengan biaya lebih rendah daripada desain menggunakan Pd T-05-2005 yang telah dimodifikasi menjadi pedoman interim No.002/P/BM/2011.



Gambar 3.2 Solusi Overlay Berdasarkan Lendutan Balik Benkelman Beam untuk WMAPT 41°C

(Sumber: Direktorat Jenderal Bina Marga, 2017)

3.2.3 Tebal Overlay Berdasarkan Lengkung Lendutan

Lengkung lendutan digunakan untuk perkerasan dengan beban lalu lintas desain lebih besar dari 100.000 ESA4. Lengkung lendutan dinyatakan pada titik belok lengkungan atau CF (*curvature function*) berdasarkan bentuk lengkung lendutan sebagai berikut:

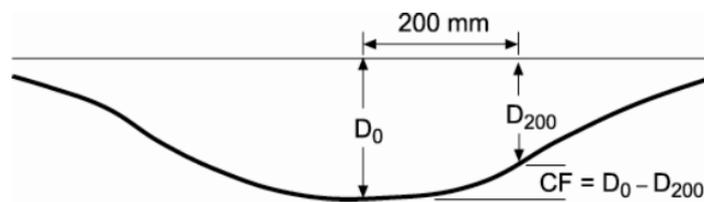
$$CF = D_0 - D_{200} \quad (3.3)$$

Keterangan:

D_0 = Lendutan maksimum pada suatu titik uji (mm);

D_{200} = Lendutan yang terjadi pada titik yang berjarak 200 mm dari titik uji tersebut (mm).

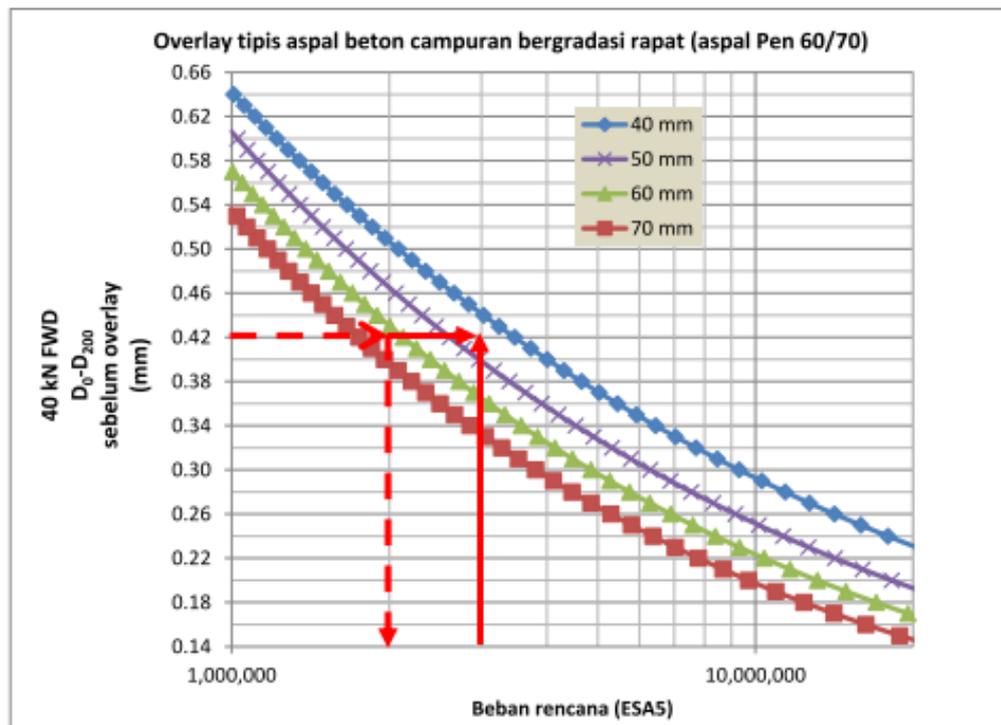
Gambar 3.3. menunjukkan skema dimensi fungsi lengkung lendutan (*curvature function* atau titik belok).



Gambar 3.3 Fungsi Lengkung Lendutan

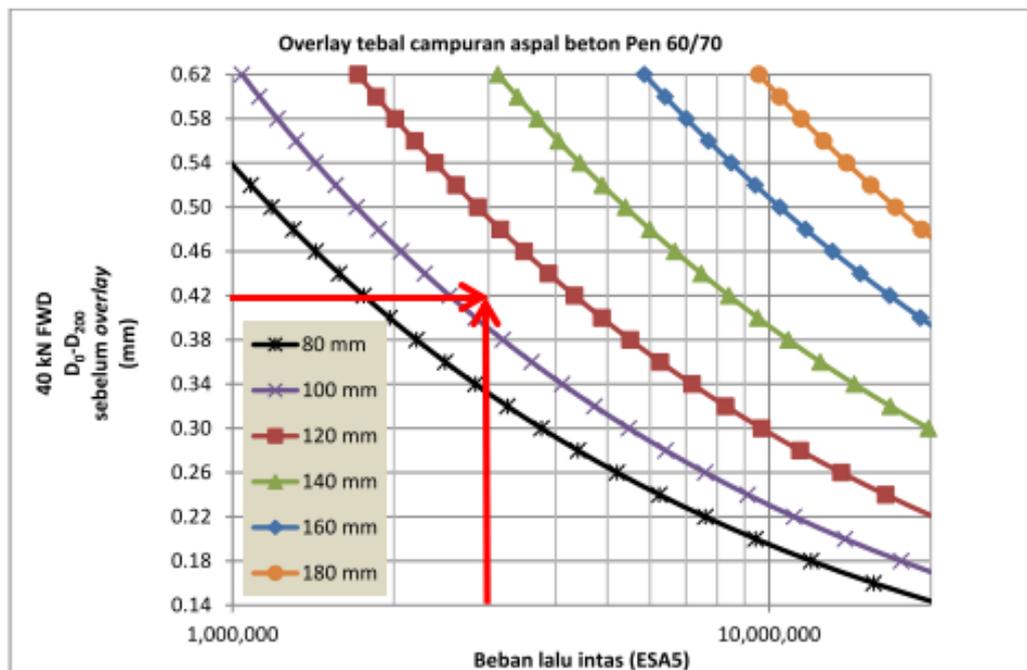
(Sumber: Austroads, 2008)

Tebal overlay berdasarkan lengkung lendutan dapat ditentukan berdasarkan *overlay* tipis atau *overlay* tebal seperti ditunjukkan pada Gambar 3.4. dan 3.5. dengan contoh penggunaan seperti diuraikan di bawah ini:



Gambar 3.4 Tebal *Overlay* Aspal Konvensional untuk Mencegah Retak Akibat Lelah pada $MAPT > 35^\circ C$ (*Overlay* tipis)

(Sumber: Direktorat Jenderal Bina Marga, 2017)



Gambar 3.5 Tebal *Overlay* Aspal Konvensional untuk Mencegah Retak Akibat Lelah pada MAPT > 35° C (*Overlay* tebal)
(Sumber: Direktorat Jenderal Bina Marga, 2017)

3.2.4 Penyesuaian Nilai Pengukuran Lendutan Terhadap Musim

Besar lendutan permukaan perkerasan aspal dipengaruhi oleh jenis tanah dan kelembaban tanah dasar. Selain dari ketinggian muka air tanah, kelembaban tanah dasar dipengaruhi oleh iklim. Atas pertimbangan tersebut maka pengukuran sebaiknya dilakukan pada waktu perkerasan dalam keadaan terlemah yaitu pada musim penghujan.

Apabila survei lendutan dilaksanakan pada musim kemarau maka nilai lendutan harus dikoreksi. Faktor koreksi terhadap musim adalah angka perbandingan antara lendutan maksimum pada musim penghujan dan lendutan pada musim kemarau:

Faktor koreksi musim kemarau = 1,2

Faktor koreksi musim penghujan = 1,0*

*Berlaku untuk pengujian lendutan pada musim penghujan atau jika muka air tanah lebih dalam dari 3 m di bawah level tanah dasar.

3.2.5 Penyesuaian Nilai Pengukuran Lendutan Terhadap Temperatur Pengujian

Untuk *overlay* diatas perkerasan berbutir, hasil pengukuran lendutan perlu dikoreksi. Hal ini dikarenakan temperatur perkerasan mempengaruhi kekakuan dan kinerja perkerasan beraspal dalam merespon beban. Apabila temperatur perkerasan pada saat pengukuran dan pada kondisi pelayanan berbeda secara signifikan, lengkung lendutan yang diukur tidak mewakili respon perkerasan terhadap pembebanan lalu lintas. Untuk itu, diperlukan faktor koreksi temperatur.

Secara umum, temperatur perkerasan tahunan rata-rata di Indonesia adalah 42⁰C pada daerah pesisir dan 38⁰C pada daerah pegunungan. Temperatur perkerasan rata-rata 41⁰C digunakan sebagai acuan dalam Manual Perkerasan Jalan Nomor 04/SE/Db/2017.

Faktor koreksi temperatur untuk pengukuran lendutan dihitung mengikuti prosedur berikut.

1. Tentukan faktor koreksi temperatur, f_T , sebagai berikut

$$f_T = \frac{\text{MAPT}}{\text{Temperatur Perkerasan Saat Pengukuran Lendutan}} \quad (3.4)$$

2. Tentukan faktor koreksi temperatur menggunakan Tabel 3.6 dan 3.7 apabila pengujian lendutan menggunakan *Benkelman Beam*. Pengujian lendutan perkerasan dengan tebal permukaan beraspal kurang dari 25 mm tidak memerlukan faktor koreksi temperatur.

Tabel 3.6 Faktor Koreksi Temperatur Lendutan (D_0) untuk *Benkelman Beam*

<u>AMPT</u> TEMP _{lapangan}	Tebal Aspal Eksisting (mm)						
	25	50	100	150	200	250	300
0,50	0,94	0,90	0,84	0,78	0,74	0,67	0,58
0,60	0,95	0,92	0,86	0,81	0,77	0,70	0,62
0,70	0,96	0,94	0,89	0,85	0,81	0,75	0,69
0,80	0,97	0,96	0,92	0,90	0,87	0,82	0,78
0,90	0,99	0,98	0,96	0,95	0,93	0,90	0,88
1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00
1,10	1,01	1,01	1,03	1,03	1,05	1,06	1,08
1,20	1,02	1,02	1,05	1,07	1,10	1,14	1,16
1,30	1,03	1,04	1,06	1,10	1,14	1,20	1,24
1,40	1,03	1,05	1,08	1,12	1,18	1,26	1,31
1,50	1,04	1,06	1,09	1,14	1,21	1,31	1,37
1,60	1,04	1,07	1,11	1,16	1,25	1,37	1,42
1,70	1,04	1,08	1,12	1,20	1,30	1,38	1,50
1,80	1,04	1,09	1,13	1,22	1,35	1,37	1,55

Sumber: Direktorat Jenderal Bina Marga, 2017

Tabel 3.7 Faktor Koreksi Temperatur Lendutan (D_0) untuk *Benkelman Beam*

<u>AMPT</u> TEMP _{lapangan}	Tebal Aspal Eksisting (mm)							
	25	50	75	100	150	200	250	300
0,50	0,93	0,81	0,72	0,64	0,54	0,51	0,48	0,43
0,60	0,95	0,85	0,77	0,72	0,64	0,58	0,53	0,48
0,70	0,96	0,89	0,83	0,79	0,73	0,66	0,61	0,57
0,80	0,98	0,92	0,88	0,85	0,81	0,76	0,72	0,69
0,90	0,99	0,96	0,93	0,92	0,89	0,86	0,84	0,83
1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00
1,10	1,01	1,02	1,03	1,03	1,04	1,06	1,09	1,11

**Lanjutan Tabel 3.7 Faktor Koreksi Temperatur Lendutan (D_0) untuk
*Benkelman Beam***

AMPT TEMP_{lapangan}	Tebal Aspal Eksisting (mm)							
	25	50	75	100	150	200	250	300
1,20	1,02	1,04	1,07	1,08	1,11	1,16	1,20	1,24
1,30	1,03	1,07	1,11	1,13	1,18	1,24	1,31	1,36
1,40	1,04	1,09	1,14	1,18	1,24	1,32	1,41	1,46
1,50	1,05	1,11	1,17	1,22	1,29	1,39	1,49	1,56
1,60	1,06	1,13	1,20	1,26	1,35	1,44	1,57	1,64
1,70	1,07	1,14	1,23	1,29	1,39	1,50	1,64	1,71
1,80	1,07	1,16	1,25	1,32	1,44	1,55	1,70	1,78

Sumber: Direktorat Jenderal Bina Marga, 2017

3.2.6 Penyesuaian Nilai Pengukuran Lendutan dan Lengkung Lendutan

Lendutan dan lengkung lendutan yang diuji menggunakan Benkelman Beam dan *FWD* akan menghasilkan nilai yang berbeda, oleh karena itu diperlukan penyesuaian atau standarisasi hasil pengukuran. Tabel 3.8 merupakan Faktor Penyesuai Lengkung Lendutan ($D_0 - D_{200}$) *BB* ke *FWD* dan Tabel 3.9 merupakan Faktor Penyesuai Lengkung Lendutan (D_0) *FWD* ke *BB*.

Tabel 3.8 Faktor Penyesuaian Lengkung Lendutan ($D_0 - D_{200}$) *BB* ke *FWD*

Tebal Aspal Eksisting (mm)	Faktor	Tebal Aspal Eksisting (mm)	Faktor
0	1,00	160	0,69
20	0,95	180	0,67
40	0,91	200	0,65
60	0,86	220	0,63
80	0,82	240	0,61
100	0,79	260	0,60
120	0,75	280	0,59
140	0,72	300	0,59

Sumber: Direktorat Jenderal Bina Marga, 2017

Tabel 3.9 Faktor Penyesuaian Lendutan (D_0) FWD ke BB

Tebal Aspal Eksisting (mm)	Faktor	Tebal Asapl Eksisting (mm)	Faktor
0	1,00	160	1,26
20	1,12	180	1,28
40	1,14	200	1,29
60	1,16	220	1,31
80	1,18	240	1,33
100	1,20	260	1,34
120	1,22	280	1,34
140	1,24	300	1,36

Sumber: Direktorat Jenderal Bina Marga, 2017

3.3 Visual Basic 6.0

3.3.1 Pengertian

Kata “visual” menunjukkan metode dalam merancang *Grapical user interface* (GUI), yaitu cukup memasukkan objek-objek yang sudah jadi ke layar dan pengaturannya langsung dilakukan dilayar itu juga secara visual, bukan dengan memakai baris-baris kode program yang jumlahnya bisa mencapai ribuan baris.

Sedangkan kata “Basic” merupakan singkatan dari “*Beginners All-Purpose Symbolic Intruction code*”. Basic juga merupakan program berbasis bahasa program yang sebelumnya masih dalam versi *DOS*, dan merupakan salah satu bahasa pemograman yang tertua dalam sejarah komputer. Salah satu faktor yang menyebabkan penulis memilih bahasa *visual basic* adalah kemampuannya dalam menangani jumlah variabel yang sangat besar.

Menurut Kusrini (2007), *visual basic* adalah salah satu bahasa pemograman komputer. Bahasa pemograman adalah perintah-perintah yang dimengerti oleh komputer untuk melakukan tugas-tugas tertentu. *Visual basic* merupakan salah satu *development tool* yaitu alat bantu untuk membuat berbagai macam program komputer khususnya yang menggunakan sistem operasi windows.

Versi awal *visual basic* diciptakan oleh Alan Cooper dan kemudian menjualnya ke *Microsoft* yang mengambil alih pengembangan produk dengan

memberi nama sandi “*Thunder*”. Akhirnya *Visual basic* menjadi bahasa pemrograman utama di lingkungan *Windows*. Sebelum mencapai kepopuleran seperti sekarang, *Visual basic* telah dirilis beberapa kali seperti berikut:

1. Versi 1.0 dirilis bulan Mei 1991.
2. Januari 1992 dilengkapi control custom (*Professional Toolkit*).
3. Versi 2.0 yang lebih “canggih” dirilis pada bulan Oktober 1992.
4. Versi 3.0 dirilis tahun 1993 dengan “*Development Tool*” andal.
5. Versi 4.0 dirilis tahun 1995 dilengkapi *Object Oriented Programming*, mulai dipakai untuk mengembangkan aplikasi berbasis *Windows*.
6. Versi 5.0 dirilis tahun 1997 dengan *Integrated Development Environment (IDE)*.
7. Versi 6.0 dirilis tahun 1998 bersama dengan *Microsoft Visual Studio*.
8. Versi .NET dirilis tahun 2001, yaitu *Visual basic.NET* bersama dengan *Microsoft Visual Studio.NET*.

3.3.2 Keunggulan

Visual basic mempunyai banyak kelebihan dibandingkan Perangkat lunak / Bahasa pemrograman yang lain. Dibawah ini adalah beberapa keunggulan tersebut:

1. Kurva pembelajaran dan pengembangan yang lebih singkat dibandingkan bahasa pemrograman yang lain seperti C/C++, *Delphi* atau bahkan *PowerBuilder* sekalipun.
2. Menghilangkan kompleksitas pemanggilan fungsi *Windows API*, karena banyak fungsi-fungsi tersebut sudah di-*embedded* ke dalam *syntax Visual basic*.
3. Cocok digunakan untuk mengembangkan aplikasi/program yang bersifat “*Rapid Application Development*”.
4. Juga sangat cocok digunakan untuk membuat program / aplikasi Bisnis.
5. Digunakan oleh hampir semua keluarga *Microsoft Office* sebagai Bahasa *Macro*-nya, segera akan diikuti oleh yang lain.
6. Dapat membuat *ActiveX Control*.
7. Dapat menggunakan OCX / Komponen yang disediakan oleh pihak ketiga (*third party*) sebagai *tool* pengembangan.

8. Menyediakan wizard yang sangat berguna untuk mempersingkat / mempermudah pengembangan aplikasi.
9. Mendekati *Object Oriented Programming*.
10. Dapat di-integrasikan dengan Internet, baik itu pada sisi *Client* maupun pada sisi *Server*.
11. Dapat membuat *ActiveX Automation Server*.
12. Integrasi dengan *Microsoft Transaction Server*.

3.3.3 Konsep Dasar Pemrograman dalam *Visual basic 6.0*

Konsep dasar pemrograman *Visual basic 6.0* adalah pembuatan form dengan mengikuti aturan pemrograman *Property*, *Metode* dan *Event*. Hal ini berarti:

1. *Property*: Setiap komponen di dalam pemrograman *Visual basic* dapat diatur propertinya sesuai dengan kebutuhan aplikasi. *Property* yang tidak boleh dilupakan pada setiap komponen adalah “*Name*”, yang berarti nama variabel (komponen) yang akan digunakan dalam *scripting*. Properti “*Name*” ini hanya bisa diatur melalui jendela *Property*, sedangkan nilai peroperti yang lain bisa diatur melalui *script* seperti

```
Command1.Caption="Play"
```

```
Text1.Text="Visual basic"
```

```
Label1.Visible=False
```

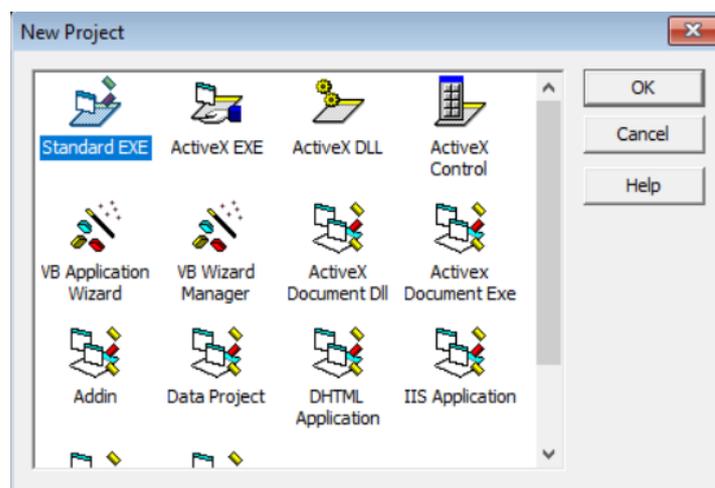
```
Timer1.Enable=True
```

2. *Metode*: Bahwa jalannya program dapat diatur sesuai aplikasi dengan menggunakan metode pemrograman yang diatur sebagai aksi dari setiap komponen. Metode inilah tempat untuk mengekspresikan logika pemrograman dari pembuatan suatu program aplikasi.
3. *Event*: Setiap komponen dapat beraksi melalui event, seperti event click pada command button yang tertulis dalam layar script *Command1_Click*, atau *event Mouse Down* pada *picture* yang tertulis dengan *Picture1_MouseDown*. Pengaturan event dalam setiap komponen yang akan menjalankan semua metode yang dibuat.

3.3.4 Komponen Dalam Membuat Project Baru Pada *Visual basic 6.0*

Adapun Komponen – Komponen untuk membangun program pada *Visual basic* diantaranya sebagai berikut:

3. Untuk memulai pembuatan program aplikasi di dalam *Visual basic*, yang dilakukan adalah membuat project baru. Project adalah sekumpulan form, modul, fungsi, data dan laporan yang digunakan dalam suatu aplikasi. Membuat project baru dapat dilakukan dengan memilih menu [*File*] >> [*New Project*] atau dengan menekan ikon  [*new project*] pada Toolbar yang terletak di pojok kiri atas. Setelah itu akan muncul konfirmasi untuk jenis project dari program aplikasi yang akan dibuat seperti terlihat pada gambar 3.6. berikut.



Gambar 3.6 Layar Pemilihan jenis *project*

(Sumber: Screen picture *Visual basic*, diakses Tahun 2017)

Visual basic 6.0 menyediakan 13 jenis project yang bisa dibuat seperti terlihat pada gambar 3.6 di atas. Ada beberapa project yang biasa digunakan oleh banyak pengguna *Visual basic*, antara lain:

- a. *Standard EXE*: Project standar dalam *Visual basic* dengan komponen-komponen standar. Jenis project ini sangat sederhana, tetapi memiliki keunggulan bahwa semua komponennya dapat diakui oleh semua unit komputer dan semua user meskipun bukan administrator.

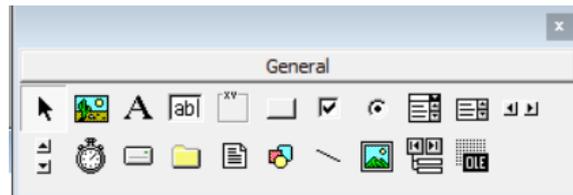
- b. *ActiveX EXE*: Project ini adalah project ActiveX berisi komponen-komponen kemampuan untuk berinteraksi dengan semua aplikasi di sistem operasi windows.
 - c. *ActiveX DLL*: Project ini menghasilkan sebuah aplikasi *library* yang selanjutnya dapat digunakan oleh semua aplikasi di sistem operasi windows.
 - d. *ActiveX Control*: Project ini menghasilkan komponen-komponen baru untuk *visual basic* yang lain.
 - e. *VB Application Wizard*: Project ini memandu pengguna untuk membuat aplikasi secara mudah tanpa harus pusing-pusing dengan perintah-perintah pemrograman.
 - f. *Addin*: Project seperti *Standard EXE* tetapi dengan berbagai macam komponen tambahan yang memungkinkan kebebasan kreasi dari pengguna.
 - g. *Data project*: Project ini melengkapi komponennya dengan komponen-komponen database. Sehingga bisa dikatakan project ini memang disediakan untuk keperluan pembuatan aplikasi database.
 - h. *DHTML Application*: Project ini digunakan untuk membuat aplikasi internet pada sisi client (*client side*) dengan fungsi-fungsi *DHTML*.
 - i. *IIS Application*: Project ini menghasilkan aplikasi internet pada sisi server (*server side*) dengan komponen-komponen CGI (*Common Gateway Interface*).
4. *Toolbar* adalah tombol – tombol yang mewakili suatu perintah tertentu dari *Visual basic*. Setiap tombol tersebut dapat langsung diklik untuk melakukan perintah tertentu. Biasanya tombol-tombol ini merupakan perintah-perintah yang sering digunakan dan terdapat pula pada menu *Visual basic*. Gambar 3.7 merupakan gambar dari *Toolbar* standar *Visual basic*



Gambar 3.7 Toolbar standard Visual basic

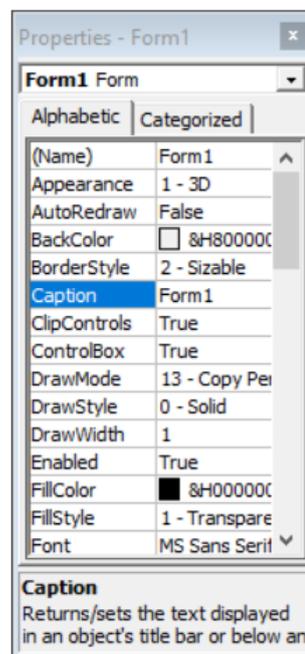
(Sumber: Screen picture *Visual basic*, diakses Tahun 2017)

5. *Toolbox* adalah sebuah kotak perintah yang mengandung semua objek atau kontrol yang dibutuhkan untuk membentuk suatu program aplikasi. Kontrol adalah suatu objek yang akan menjadi interface (penghubung) antara program aplikasi dan user-nya, dan kesemuanya harus diletakkan didalam jendela form diatas. Berikut adalah gambar *Toolbox* yang dapat dilihat pada Gambar 3.8.



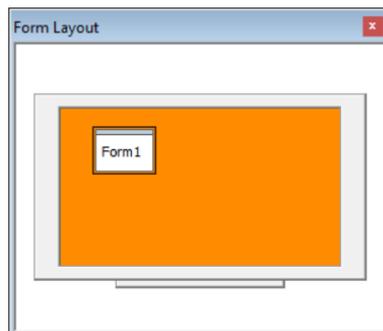
Gambar 3.8 *Toolbox* standard *Visual basic*
(Sumber: Screen picture *Visual basic*, diakses Tahun 2017)

4. Jendela properties adalah jendela yang mengandung semua informasi mengenai objek yang terdapat pada aplikasi *Visual basic*. Properti adalah sifat dari sebuah objek, misalnya seperti namanya, warna, ukuran, posisi, dan sebagainya. Setiap objek sebagian besar memiliki jenis properti yang sama, tetapi adapula yang berbeda-beda. Berikut adalah gambar Jendela Propertis yang dapat dilihat pada Gambar 3.9.



Gambar 3.9 Jendela Properties
(Sumber: Screen picture *Visual basic*, diakses Tahun 2017)

5. *Form Layout Window* adalah jendela yang menggambarkan posisi dan form yang ditampilkan pada layar monitor. Posisi *form* pada *Form Layout Window* inilah yang merupakan petunjuk dimana aplikasi akan ditampilkan pada layar monitor saat dijalankan nanti. Berikut adalah gambar *Form Layout Window* yang dapat dilihat pada Gambar 3.10.



Gambar 3.10 *Form Layout Window*

(Sumber: Screen picture *Visual basic*, diakses Tahun 2017)

3.4 Penyetelan Kurva (*Curve Fitting*)

3.4.1 Tinjauan Umum

Data-data yang bersifat diskrit dapat dibuat *continuum* melalui proses *curve fitting*. *Curve fitting* merupakan proses data *smoothing*, yakni proses pendekatan terhadap kecenderungan data-data dalam bentuk persamaan model matematika., harga fungsi $f(x)$ di dalam rentang data dapat diperkirakan. Bila nilai $f(x)$ yang dicari cukup banyak, maka akan lebih prektis bila sebelum menentukan $f(x)$ diperkirakan dulu hubungan matematika antara x dan $f(x)$ dari data yang tersedia. Permasalahan ini disebut penyetelan kurva atau regresi.

Jadi penyetelan kurva (*curve fitting*) bertujuan untuk mencari kurva terbaik (*insomesense* atau dalam kriteria tertentu) yang mampu merepresentasikan/menggambarkan hubungan matematika antara x dan $f(x)$. ukuran/kriteria untuk mengukur kualitas kurva terstel dapat diekspresikan dengan berbagai cara ukuran penyimpangan/kesalahan (*error*).

Kesalahan/penyimpangan data ditentukan berdasarkan kesalahan mutlak (*absolute error*) atau kesalahan kuadrat terkecil (*least square error*).

Kesalahan berdasarkan nilai mutlak:

$$\varepsilon = \sum_{i=1}^m |y_i - \bar{y}(\underline{a})| \quad (3.5)$$

keterangan:

m = jumlah data

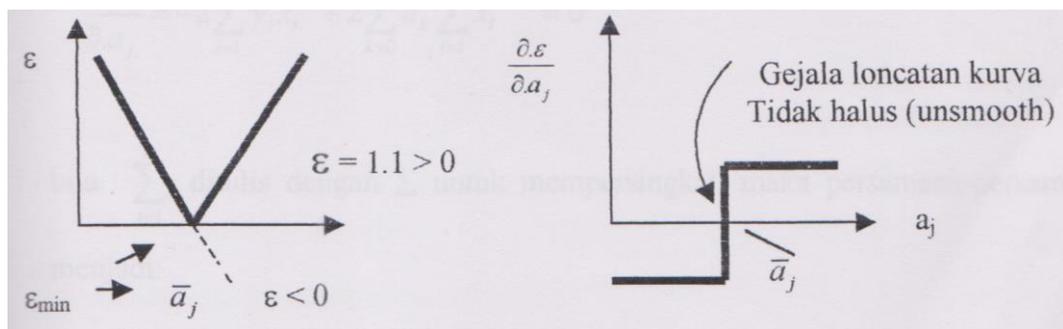
$\underline{a} = a_j$, j = indeks koefisien $a_0, a_1, a_2, \dots, a_n$

Untuk meminimalkan kesalahan, maka

$$\frac{\partial \varepsilon}{\partial a_j} = 0 \quad (3.6)$$

Dimana $j = 0, 1, 2, \dots$ = indeks koefisien yang harus ditentukan

Namun dengan kriteria kesalahan absolut tersebut, nilai diferensialnya tidak diperoleh pada titik 0

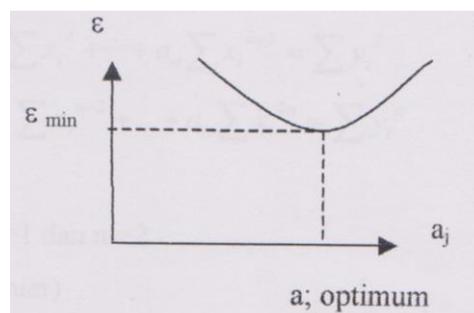


Gambar 3.11 Grafik Kriteria Kesalahan

(Sumber: Sarwidi, 2015)

Oleh karena kesulitan tersebut, kriteria kesalahan absolut jarang dipakai, yang lebih populer dipakai adalah dengan kriteria kuadrat kesalahan terkecil (*least square criterion*) sebagai berikut.

$$\varepsilon = \sum_{i=1}^m (y_i - \bar{y}(\underline{a}))^2 \quad (3.7)$$



Gambar 3.12 Grafik *Least Square Eriterion*

(Sumber: Sarwidi, 2015)

3.4.2 Penyetelan Kurva dengan Polinomial Aljabar

Polinomial aljabral atau sering disebut polynomial biasa, bahkan lazim disebut secara singkat polynomial. Untuk persamaan polynomial berorde – m dengan bentuk:

$$y = a_0 + a_1x + a_2x^2 + \dots + a_{m-1}x^{m-1} + a_mx^m \tag{3.8}$$

dapat dihasilkan persamaan-persamaan berikut ini.

$$\begin{aligned} n a_0 + a_1 \sum x_i + a_2 \sum x_i^2 + \dots + a_m \sum x_i^m &= \sum y_i \\ a_0 \sum x_i + a_1 \sum x_i^2 + a_2 \sum x_i^3 + \dots + a_m \sum x_i^{m+1} &= \sum x_i y_i \\ a_0 \sum x_i^2 + a_1 \sum x_i^3 + a_2 \sum x_i^4 + \dots + a_m \sum x_i^{m+2} &= \sum x_i^2 y_i \\ \dots & \\ \dots & \\ a_0 \sum x_i^m + a_1 \sum x_i^{m+1} + a_2 \sum x_i^{m+2} + \dots + a_m \sum x_i^{m+m} &= \sum x_i^m y_i \end{aligned} \tag{3.9}$$

atau, dapat disusun dalam bentuk perkalian matriks, sebagai berikut.

$$\begin{bmatrix} n & \sum x_i & \sum x_i^2 & \dots & \sum x_i^m \\ \sum x_i & \sum x_i^2 & \sum x_i^3 & \dots & \sum x_i^{m+1} \\ \sum x_i^2 & \sum x_i^3 & \sum x_i^4 & \dots & \sum x_i^{m+2} \\ \dots & \dots & \dots & \dots & \dots \\ \sum x_i^m & \sum x_i^{m+1} & \sum x_i^{m+2} & \dots & \sum x_i^{m+m} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \dots \\ a_m \end{bmatrix} = \begin{bmatrix} \sum y_i \\ \sum x_i y_i \\ \sum x_i^2 y_i \\ \dots \\ \sum x_i^m y_i \end{bmatrix} \tag{3.10}$$

$$A \quad x = \quad b$$

Dengan demikian, harga – harga $a_0, a_1, a_2, \dots, a_m$ dapat ditentukan secara simultan.