LAPORAN PENELITIAN

PENGEMBANGAN PROGRAM BANTU REALIN UNTUK PEMBELAJARAN METODE MATRIKS KEKAKUAN DENGAN FREEMAT

Oleh:

Yoyong Arfiadi

PROGRAM STUDI TEKNIK SIPIL
FAKULTAS TEKNIK
UNIVERSITAS ATMA JAYA YOGYAKARTA
2013

HALAMAN PENGESAHAN

1	Judul Penelitian	:	PENGEMBANGAN PROGRAM BANTU REALIN UNTUK PEMBELAJARAN METODE MATRIKS KEKAKUAN DENGAN FREEMAT
2	Peneliti Utama		
	a. Nama Lengkap dan Gelar	:	Prof. Ir. Yoyong Arfiadi, M.Eng., Ph.D.
	b. Jenis Kelamin	:	Laki-laki
	c. NPP	:	07.88.273
	d. Pangkat, Golongan	:	Profesor, IV-D
	e. Jabatan Struktural	:	-
	f. Jabatan Fungsional	:	Dosen Tetap
	g. Fakultas/Jurusan	:	Teknik/Teknik Sipil
	h. Pusat Penelitian	:	Universitas Atma Jaya Yogyakarta
	i. Alamat	:	Jl. Babarsari 44,
	j. Telpon/Faks.	:	0274-487711, 0274-487748 :
	k. Alamat Rumah	:	Jl. Cemara D-75, Sinduharjo, Ngaglik Sleman, Yogyakarta 55581
	l. Telpon/Faks	:	0274-864455
	m. e-mail	:	yoyong@mail.uajy.ac.id
3	Usul Jangka Waktu Penelitian	:	6 (enam) bulan
4	Pembiayaan		
	a. Usul Biaya	:	Rp. 3.994.000,-

Yogyakarta, Agustus 2013

Mengetahui,
Dekan ,

Ketua Peneliti,

Dr. Ir. A.M. Ade Lisantono, M.Eng.
NPP. 01.88.265

Prof. Ir. Yoyong Arfiadi, M.Eng., Ph.D.
NPP. 07.88.273

Menyetujui,
Ketua Lembaga Penelitian & Pengabdian Masyarakat UAJY,

Dr. Ir. Y. Djarot Purbadi, MT.
NPP. 07.87.217

DAFTAR ISI

BAB I
PENDAHULUAN

1.1. TINJAUAN UMUM

Penggunaan komputer dewasa ini telah menjadi bagian utama dalam kehidupan modern. Dalam bidang mekanika struktur, sudah menjadi tidak masalah lagi untuk melakukan analisis struktur dengan ribuan derajat kebebasan. Hal ini terjadi karena kemajuan yang sangat pesat, baik dalam bidang perangkat lunak maupun perangkat keras. Beberapa program analisis struktur yang banyak digunakan saat ini di antaranya adalah ETABS Nonlinear, SAP 2000, dan STAAD III. Program-program tersebut sangat mudah digunakan dengan menu yang interaktif. Untuk menggunakan program-program tersebut pengguna tinggal mengikuti petunjuk dari buku manual yang disediakan oleh setiap program tersebut. Setelah semua data isian disiapkan pengguna hanya tinggal meng'klik' tombol 'analisis' untuk mendapatkan hasilnya. Tampak bahwa program tersebut bertindak sebagai *black box* dan tentunya prinsip GIGO (*garbage in garbage out*) berlaku di sini. Dari pengalaman, tampak bahwa walaupun mahasiswa telah memperoleh keluaran dari program yang sangat canggih tersebut sejumlah mahasiswa bahkan tidak mengetahui 'harus diapakan dengan *output* tersebut'. Hal ini terjadi karena mahasiswa tersebut kurang memahami langkah-langkah yang dilakukan oleh program yang mereka gunakan. Tampak bahwa penggunaan program komputer oleh orang yang bukan ahli merupakan suatu bahaya yang sangat besar. Selain itu perlu diingat bahwa program yang canggih tersebut tidak membuat seseorang pengguna menjadi seorang *engineer* yang hebat dalam sesaat.

Hal yang telah diungkapkan di atas merupakan suatu permasalahan dalam pembelajaran analisis struktur terutama untuk bahasan metode matriks kekakuan. Di satu sisi, dengan pesatnya arus perkembangan dalam bidang teknologi informasi, penggunaan komputer harus menjadi kebiasaan dalam segala aspek kehidupan; termasuk untuk melakukan analisis struktur. Namun penggunaan komputer yang tidak pada tempatnya akan merupakan suatu bencana. Masalah ini lebih dipertegas lagi karena metode matriks kekakuan saat ini telah diterima sebagai metode standar untuk menganalisis struktur (Balfour, 1986, Krishnamoorthy, 1987). Karena dalam proses hitungan dengan metode matriks kekakuan melibatkan operasi matriks yang cukup banyak, maka tidaklah bijaksana apabila kita mengharuskan mahasiswa melakukan perkalian atau penambahan elemen matriks yang berulang-ulang. Selain hal ini akan membosankan, sehingga metode matriks kekakuan menjadi topik yang tidak menarik, juga hal ini akan menghilangkan esensi dari pemahaman langkah-langkah yang harus dikerjakan. Perlu diingat juga bahwa kita tidak mengajarkan cara menghitung tetapi mengajarkan bagaimana memecahkan persoalan dalam analisis struktur.

Beberapa peneliti telah mengembangkan program analisis struktur untuk tujuan pengajaran di lingkungan perguruan tinggi. Program ini memang tidak secanggih program komersial yang ada, namun dengan program ini pengguna terlibat langsung dengan langkah yang harus dilakukan. Wilson (1979, 1986) mengembangkan program CAL yang banyak digunakan di banyak perguruan tinggi terkemuka. Serupa dengan Wilson, Shim (1980), dan Kanok-Nukulchai (1993) telah melakukan pengembangan program bantu untuk analisis struktur AIT (*Analysis Interpretive Treatise*). Namun demikian masih terdapat kelemahan pada program-program tersebut, terutama karena kurang nyaman dipakai dan keharusan menggunakan format tertentu pada masukan datanya.

Menyadari kekurangan yang ada pada program-program terdahulu, Arfiadi (1997) telah mengembangkan program bantu analisis struktur dalam Matlab yang mudah digunakan tetapi tetap memperhatikan tujuan agar mahasiswa memahami langkah-langkah dalam metode matriks kekakuan. Dengan menggunakan subprogram yang telah dikembangkan pengguna dapat dengan mudah melakukan analisis struktur dengan metode matriks kekakuan. Namun, langkah-langkah yang dilakukan harus ditentukan oleh pengguna. Dengan demikian program ini sangat cocok untuk digunakan dalam pengajaran di kelas karena dapat menghindari operasi matriks dalam ukuran besar tetapi masih bisa menggambarkan proses analisis struktur secara keseluruhan.

Mengingat Matlab (Mathworks, 1996) merupakan program komersial dan berharga cukup mahal, dirasa ada kendala dalam penggunaannya di kalangan perguruan tinggi, terutama bagi kalangan mahasiswa. Saat ini terdapat beberapa *open source* yang mempunyai kemampuan yang sama dengan Matlab di antaranya adalah FreeMat (www.freemat.org), Octave (www.octave.org) dan Scilab (www.scilab.org). Dalam penelitian ini program FreeMat (Basu, 2011) digunakan karena lebih *user friendly* dan hampir menyerupai Matlab. Dengan digunakannya program ini, maka kalangan perguruan tinggi tidak perlu lagi mengeluarkan beaya yang banyak untuk pembelajaran metode kekakuan di lingkungan kampus mereka. Dengan dikembangkannya program bantu dalam penelitian ini maka terdapat beberapa keuntungan sebagai berikut:

a) pengajaran metode matriks kekakuan menjadi lebih menarik,
b) persoalan yang dibahas tidak terlalu dibatasi dengan jumlah derajat kebebasan seperti pada hitungan secara manual (dengan kalkulator),
c) tersedia program analisis struktur dengan metode kekakuan tanpa mengeluarkan beaya, dan
d) mahasiswa dapat memahami langkah dalam analisis struktur dengan metode kekakuan dibandingkan dengan menggunakan program *black box* seperti program SAP dan ETABS.

Dalam penelitian ini akan dikembangkan program-program untuk analisis struktur rangka bidang (*plane truss*) dan struktur portal bidang (*plane frame*). Pada struktur portal bidang akan dikembangkan pula hal-hal khusus seperti batang dengan ujung sendi, pengaruh deformasi geser dan pengaruh daerah kaku pada ujung batang. Hal khusus ini merupakan pengembangan dari program yang telah dikembangkan sebelumnya dalam Matlab (Arfiadi, 1997).

1.2. FORMULASI MASALAH

Analisis struktur dengan metode matriks kekakuan saat ini merupakan suatu analisis standar sebagai hal dasar yang harus dipunyai seorang sarjana teknik sipil. Karena melibatkan operasi matriks yang kadang-kadang cukup besar ukurannya, menyebabkan pembelajaran metode matriks kekakuan tidak mudah diajarkan di kelas. Hal ini lebih dikuatkan lagi dengan tersedianya perangkat lunak komersial yang dapat menyediakan hasil analisis secara langsung dari masukan (input) yang diberikan. Karena dalam program komersial proses hitungan tersembunyi dalam program, maka pengguna hanya memakai program tersebut sebagai *black box* tanpa mengetahui proses hitungan yang ada. Untuk menjembatani hal itu, perlu dibuat program komputer di mana mahasiswa dapat berinteraksi selama proses berlangsung. Selain itu proses dan langkah penyelesaian ditentukan oleh mahasiswa sendiri. Untuk itu diperlukan subprogram-subprogram untuk mengatasi hal ini. Program yang dikembangkan harus mudah diperoleh dengan beaya yang tidak terlalu besar, dan jika perlu tanpa beaya sama sekali.

1.3. TINJAUAN PUSTAKA

Dalam analisis struktur dikenal dua metode yaitu metode klasik dan metode matriks. Metode klasik seperti metode *slope deflection*, metode Cross diperuntukkan struktur tertentu dan ditujukan untuk penyelesaian secara manual dengan kalkulator. Metode matriks merupakan metode yang lebih terstruktur dan modular, sehingga dapat digunakan untuk penyelesaian yang lebih umum dan mudah diprogram dengan komputer. Metode matriks kekakuan sebenarnya telah lama berkembang (lihat Kassimali, 2012). Di antara yang telah melakukan pengembangan metode matriks kekakuan adalah R.K. Livesley yang mempublikasikannya pada tahun 1954 (Kassimali, 2012). Setelah perkembangan komputer pribadi akhir-akhir ini, analisis struktur dengan metode matriks kekakuan sangat berkembang, yang diikuti dengan tersedianya perangkat lunak analisis struktur.

Program komputer yang digunakan umumnya adalah FORTRAN. Akhir-akhir ini bahasa pemrograman C/C++ dan Java juga telah banyak dikembangkan. Kelompok bahasa tersebut kita kelompokkan sebagai kelompok pertama.

Untuk mendukung analisis dan pemecahan suatu masalah, telah berkembang pula bahasa pemrograman seperti Mathematica dan Matlab sebagai alat bantu untuk menyelesaikan suatu persoalan melalui suatu cara yang lebih sederhana. Bahasa pemrograman ini kita sebut sebagai kelompok kedua. Dengan adanya bahasa pemrograman kelompok kedua, maka pengembangan program dapat lebih mudah dilakukan. Beberapa pemrograman yang telah dibuat dalam Matlab untuk analisis struktur dengan metode matriks kekakuan di antaranya adalah Arfiadi (1996, 2003), Austrell dkk. (2004) dan Ferreira (2006).

Saat ini telah dikembangkan suatu program yang mempunyai kemampuan analisis seperti program kelompok kedua seperti misalnya FreeMat, Scilab, Octave. Keuntungan dengan adanya program ini adalah program ini merupakan suatu *open source*, sehingga kita tidak perlu membayar untuk menggunakannnya. Mengingat umumnya mahasiswa teknik tidak membeli program sendiri, maka dengan adanya *open source* ini sangat membantu dan menguntungkan. Dalam tulisan ini dipilih FreeMat karena menu yang tersedia lebih *user friendly*, sehingga mudah digunakan sesuai dengan tujuan dari penelitian ini.

BAB II
DASAR TEORI

2.1. HUBUNGAN ANTAR VARIABEL DALAM ANALISIS STRUKTUR

Permasalahan pada metode kekakuan didasarkan pada hubungan antar variabel sebagai berikut:

(a) hubungan antara deformasi batang dan perpindahan titik-kumpul (*compatibility* atau kesepadanan),

(b) hubungan antara deformasi batang dan gaya-gaya dalam (*constitutive equation*),

(c) hubungan antara beban luar dan gaya gaya dalam (keseimbangan).

Secara skematis, hubungan-hubungan tersebut ditunjukkan pada Gambar 2.1. Berdasarkan Gambar 2.1 formulasi metode matriks kekakuan dapat diperoleh sebagai berikut ini.

(a) Dari kompatibilitas (kesepadanan) antara deformasi internal pada batang (dalam koordinat batang/lokal) dan perpindahan titik kumpul (dalam koordinat global):

$$\{u\} = [T]\{U\} \tag{2.1}$$

dengan:

$\{u\}$ = deformasi internal,

$[T]$ = matriks transformasi,

$\{U\}$ = perpindahan global titik kumpul.

(b) Dari hubungan antara gaya-gaya dalam dan deformasi internal (*constitutive law*):

$$\{S\} = [k]\{u\} \tag{2.2}$$

dengan $\{S\}$ = gaya internal batang, $[k]$ = matriks kekakuan batang (dalam koordinat lokal) dan $\{u\}$ = deformasi internal.

(c) Hubungan antara beban luar dan gaya internal (keseimbangan)

$$\{S\} = [T]\{P\} \tag{2.3}$$

dengan $\{P\}$ = vektor beban luar dalam koordinat global.

Ditulis dalam bentuk lain persamaan di atas menjadi:

$$\{P\} = [T]^{-1}\{S\} \tag{2.4}$$

Dapat dibuktikan bahwa matriks transformasi $[T]$ adalah matriks ortogonal, yaitu

$$[T]^T = [T]^{-1} \tag{2.5}$$

yang dapat dibuktikan berdasarkan prinsip

kerja luar = kerja dalam

sebagai berikut ini.

Kerja-luar: $W_E = \frac{1}{2}\{P\}^T\{U\}$

Kerja-dalam: $W_I = \frac{1}{2}\{S\}^T\{u\}$

Dari $W_E = W_I$ diperoleh:

$$\{P\}^T\{U\} = \{S\}^T\{u\}$$

Substitusikan pers. (1) pada persamaan di atas sehingga diperoleh

$$\{P\}^T\{U\} = \{S\}^T[T]\{U\}$$

atau

$$\{P\}^T = \{S\}^T[T]$$

Dengan mentransposkan persamaan tersebut diperoleh (ingat $([A][B])^T = [B]^T[A]^T$)

$$\{P\} = [T]^T[S] \tag{2.6}$$

Jika pers. (2.4) dibandingkan dengan pers. (2.6) terbukti bahwa matriks transformasi [T] adalah matriks ortogonal.

Selanjutnya substitusikan pers. (2.2) pada pers. (2.6) sehingga

$$\{P\} = [T]^T[k]\{u\}$$

dan dengan mengingat pers. (2.1) akhirnya diperoleh hubungan

$$\{P\} = [T]^T[k][T]\{U\}$$

atau

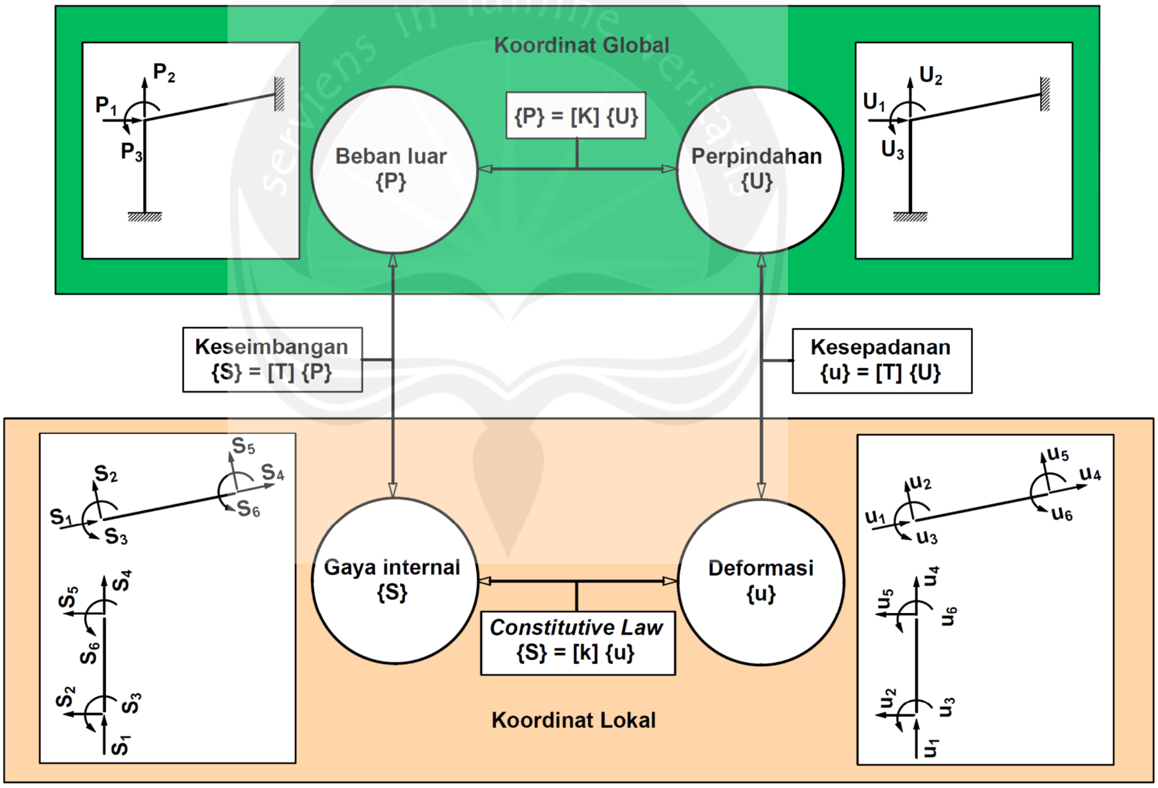
$$\{P\} = [K]\{U\}$$

dengan

$$[K] = [T]^T[k][T] \tag{2.7}$$

Dalam persamaan (2.7) matriks [K] dikenal sebagai matriks kekakuan dalam koordinat global. Selanjutnya hubungan tersebut ditunjukkan pada Gambar 2.1 dan dilakukan pada semua batang.

Formulasi ini menjadi dasar dalam analisis struktur dengan metode matriks kekakuan. Formulasi tersebut berlaku umum dan dapat digunakan baik pada struktur rangka bidang maupun portal bidang. Penggunaan dari formulasi yang dibahas dalam bab ini akan diuraikan pada bab-bab selanjutnya.



Gambar 2.1. Hubungan antar variabel dalam analisis struktur

2.2. ANALISIS STRUKTUR RANGKA BIDANG (*PLANE TRUSS*)

2.2.1. Hubungan antara gaya batang dan deformasi

Pada struktur rangka bidang hubungan antara gaya batang dan deformasi dapat dinyatakan dengan (Arfiadi, 2011, Kassimali, 2012):

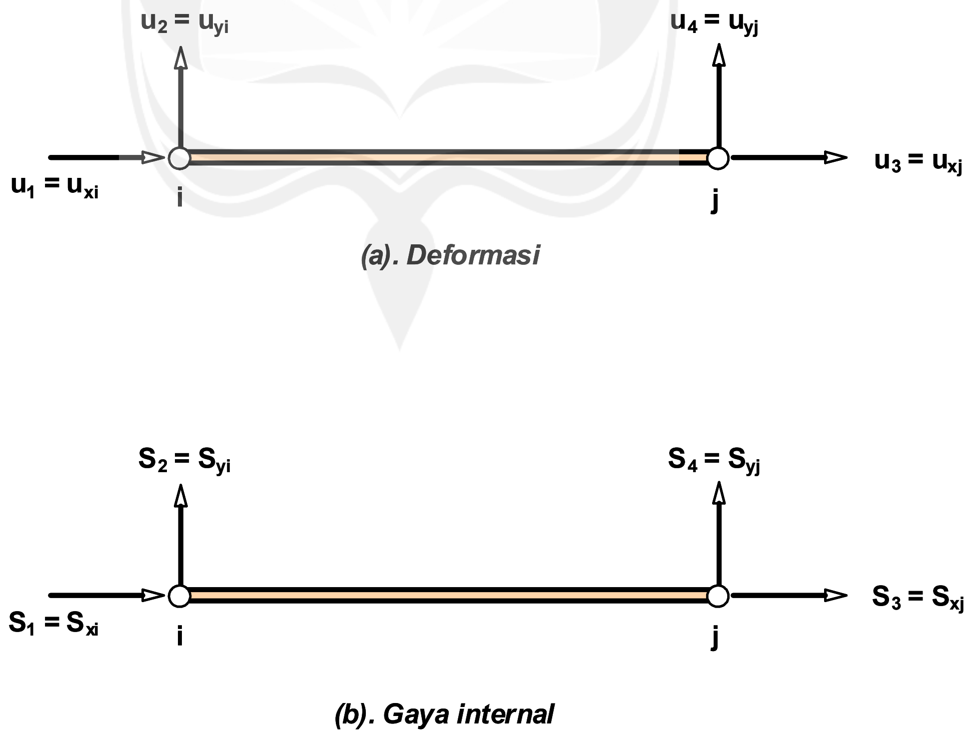
$$\{S\} = [k]\{u\} \tag{2.8a}$$

atau

$$\begin{Bmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \end{Bmatrix} = \begin{bmatrix} \frac{EA}{L} & 0 & -\frac{EA}{L} & 0 \\ 0 & 0 & 0 & 0 \\ -\frac{EA}{L} & 0 & \frac{EA}{L} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{Bmatrix} \tag{2.8b}$$

sesuai dengan Gambar 2. 2.

serviens in lumine veritatis



Gambar. 2.2. Gaya internal dan defromasi pada struktur rangka bidang

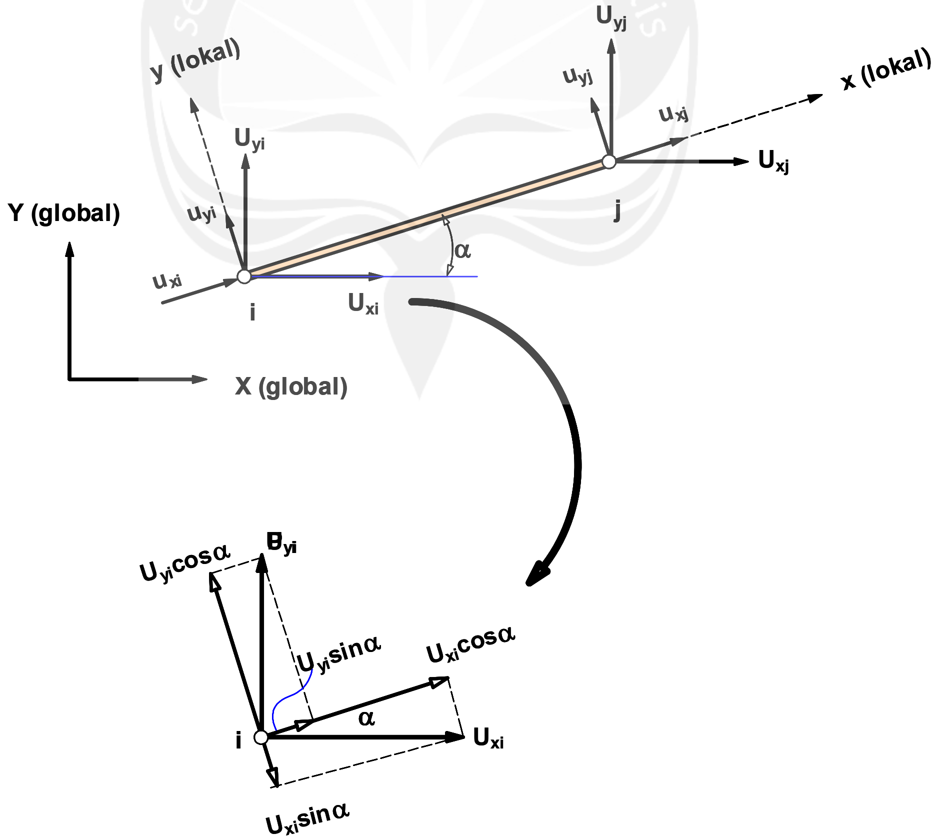
2.2.2. Hubungan antara deformasi dan perpindahan dalam sumbu global

Hubungan antara deformasi (perpindahan dalam sumbu lokal) dengan perpindahan dalam sumbu global ditunjukkan pada Gambar 2.3 dan dapat dinyatakan dengan:

$$\{u\} = [T]\{U\} \tag{2.9a}$$

atau:

$$\begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{Bmatrix} = \begin{bmatrix} \cos\alpha & \sin\alpha & 0 & 0 \\ -\sin\alpha & \cos\alpha & 0 & 0 \\ 0 & 0 & \cos\alpha & \sin\alpha \\ 0 & 0 & -\sin\alpha & \cos\alpha \end{bmatrix} \begin{Bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{Bmatrix} \tag{2.9b}$$



Gambar. 2.3. Transformasi koordinat

2.2.3. Hubungan antara gaya batang (dalam sumbu lokal) dan gaya dalam sumbu global

Dengan cara yang sama dapat diperoleh hubungan antara gaya batang {S} dalam koordinat lokal dengan gaya luar {P} dalam koordinat global, yaitu:

$$\{S\} = [T]\{P\} \quad (2.10a)$$

$$\begin{Bmatrix} S_1 \\ S_2 \\ S_{3j} \\ S_4 \end{Bmatrix} = \begin{bmatrix} \cos\alpha & \sin\alpha & 0 & 0 \\ -\sin\alpha & \cos\alpha & 0 & 0 \\ 0 & 0 & \cos\alpha & \sin\alpha \\ 0 & 0 & -\sin\alpha & \cos\alpha \end{bmatrix} \begin{Bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \end{Bmatrix} \quad (2.10b)$$

Karena $[T]$ adalah matriks bujur sangkar, maka pers. (10) dapat ditulis dalam bentuk lain sebagai

$$\{P\} = [T]^{-1}\{S\} \quad (2.11)$$

Karena matriks transformasi $[T]$ adalah matriks ortogonal di mana $[T]^{-1} = [T]^{T}$, maka pers. (11) dapat ditulis menjadi

$$\{P\} = [T]^{T}\{S\} \quad (2.12)$$

2.2.4. Matriks kekakuan dalam koordinat global

Dengan mensubstitusikan pers. (2.8), $\{S\} = [k]\{u\}$, pada pers. (12), $\{P\} = [T]^{T}\{S\}$, diperoleh

$$\{P\} = [T]^{T}\{S\}$$

$$\{P\} = [T]^{T}[k]\{u\}$$

Selanjutnya dengan mensubstitusikan pers. (2.9), $\{u\} = [T]\{U\}$, diperoleh

$$\{P\} = [T]^T [k][T]\{U\} \tag{2.13}$$

Pers. (2.13) merupakan hubungan antara gaya dalam koordinat global $\{P\}$ dan perpindahan dalam koordinat global $\{U\}$. Pers. (2.13) dapat disederhanakan menjadi

$$\{P\} = [K]\{U\} \tag{2.14}$$

dengan

$$[K] = [T]^T [k][T] \tag{2.15a}$$

$$[K] = \begin{bmatrix} \frac{EA}{L}c^2 & \frac{EA}{L}sc & -\frac{EA}{L}c^2 & -\frac{EA}{L}sc \\ \frac{EA}{L}sc & \frac{EA}{L}s^2 & -\frac{EA}{L}sc & -\frac{EA}{L}s^2 \\ -\frac{EA}{L}c^2 & -\frac{EA}{L}sc & \frac{EA}{L}c^2 & \frac{EA}{L}sc \\ -\frac{EA}{L}sc & -\frac{EA}{L}s^2 & \frac{EA}{L}sc & \frac{EA}{L}s^2 \end{bmatrix} \tag{2.15b}$$

2.2.5. Perakitan matriks kekakuan

Perakitan matriks kekakuan masing-masing batang, yang telah ditransformasikan dalam koordinat global, dilakukan dengan menggunakan vektor tujuan (Arfiadi, 2011, Balfour, 1986). Vektor tujuan merupakan nomor derajat kebebasan yang berkaitan dengan batang tersebut. Dengan cara ini maka proses perakitan dapat dilakukan dengan mudah, sehingga akhirnya diperoleh matriks kekakuan struktur, yaitu [K], sebagai:

$$[K] = \sum_{i=1}^{m} [K]_i \tag{2.16}$$

m = jumlah batang dalam struktur.

2.2.6. Vektor beban luar

Vektor beban luar selanjutnya ditentukan sesuai dengan urutan derajat kebebasan yang dipunyai struktur tersebut. Dengan ditetapkannya vector beban luar {P}, selanjutnya derajat kebebasan atau perpindahan global struktur dapat dihitung.

2.2.7. Perpindahan global

Vektor perpindahan struktur dapat diperoleh dari persamaan:

$$\{P\} = [K]\{U\} \tag{2.17}$$

Pers. (2.17) dapat diselesaikan dengan salah satu metode penyelesaian persamaan linier.

2.2.8. Perpindahan dalam koordinat lokal (deformasi batang)

Perpindahan batang dalam koordinat lokal (deformasi batang) selanjutnya dapat dihitung dari hubungan antara perpindahan sumbu lokal dan global menurut:

$$\{u\}_m = [T]_m \{U\}_m \tag{2.18}$$

Perlu dicatat bahwa $\{U\}_m$ pada pers. (12.8) merupakan bagian perpindahan global yang sesuai dengan batang yang ditinjau. Untuk struktur rangka bidang $\{U\}_m$ berukuran 4 x 1, dan disusun sesuai dengan vektor tujuan batang tersebut.

2.2.9. Gaya batang

Gaya batang selanjutnya dapat dihitung dengan persamaan:

$$\{S\}_m = [k]_m \{u\}_m \tag{2.19}$$

2.3. ANALISIS STRUKTUR PORTAL BIDANG (*PLANE FRAME*)

2.3.1. Hubungan antara gaya batang dan deformasi

Pada struktur portal bidang hubungan antara gaya batang dan deformasi dapat dinyatakan dengan (Arfiadi, 2011, Kassimali, 2012):

$$\{S\} = [k]\{u\} \tag{2.20a}$$

atau

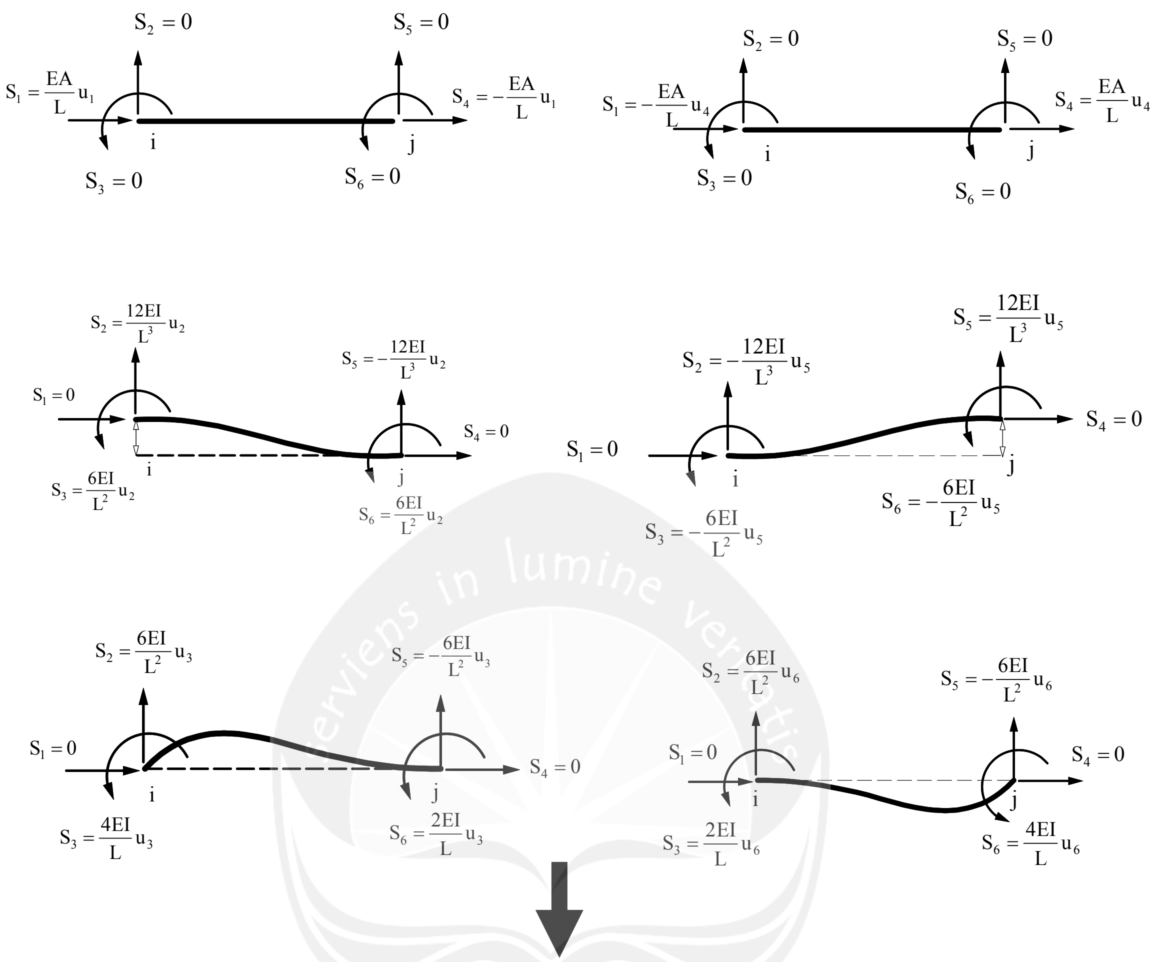
$$\begin{Bmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \\ S_5 \\ S_6 \end{Bmatrix} = \begin{bmatrix} \frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\ 0 & \frac{12EI}{L^3} & \frac{6EI}{L^2} & 0 & -\frac{12EI}{L^3} & \frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{4EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{2EI}{L} \\ -\frac{EA}{L} & 0 & 0 & \frac{EA}{L} & 0 & 0 \\ 0 & -\frac{12EI}{L^3} & -\frac{6EI}{L^2} & 0 & \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{2EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{4EI}{L} \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{Bmatrix} \tag{2.20b}$$

sesuai dengan Gambar 2.4.

2.3.2. Hubungan antara deformasi dan perpindahan dalam sumbu global

Hubungan antara deformasi (perpindahan dalam sumbu lokal) dengan perpindahan dalam sumbu global ditunjukkan ada Gambar 2. 5 dan dapat dinyatakan dengan:

$$\{u\} = [T]\{U\} \tag{2.21a}$$



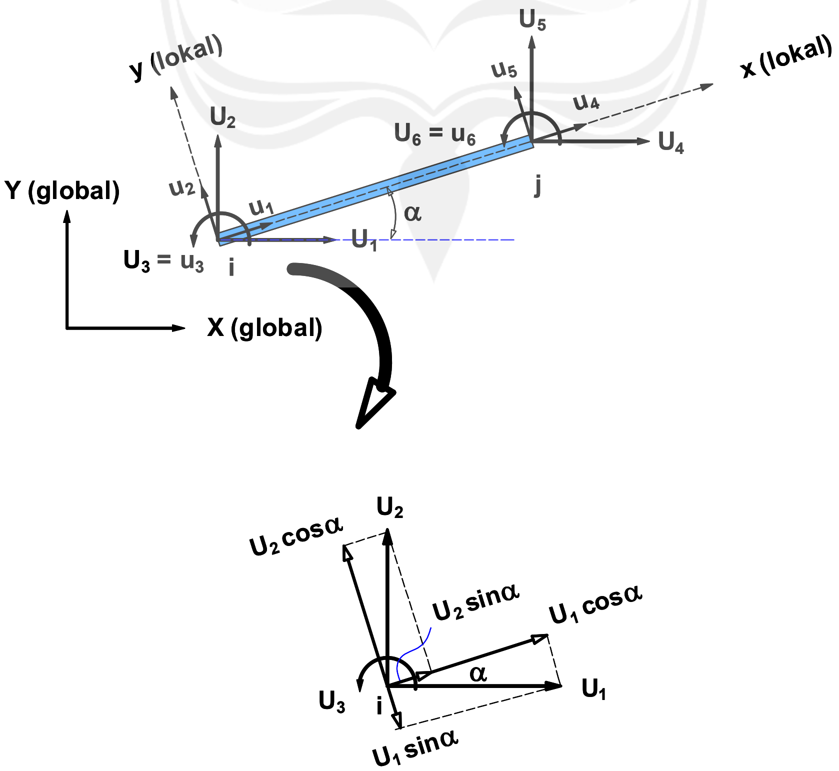
$$
[\mathbf{k}] = \begin{bmatrix}
\frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\
0 & \frac{12EI}{L^3} & \frac{6EI}{L^2} & 0 & -\frac{12EI}{L^3} & \frac{6EI}{L^2} \\
0 & \frac{6EI}{L^2} & \frac{4EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{2EI}{L} \\
-\frac{EA}{L} & 0 & 0 & \frac{EA}{L} & 0 & 0 \\
0 & -\frac{12EI}{L^3} & -\frac{6EI}{L^2} & 0 & \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\
0 & \frac{6EI}{L^2} & \frac{2EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{4EI}{L}
\end{bmatrix}
$$

Gambar 2.4. Gaya internal dan deformasi pada struktur rangka bidang atau:

$$\begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{Bmatrix} = \begin{bmatrix} c & s & 0 & 0 & 0 & 0 \\ -s & c & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & c & s & 0 \\ 0 & 0 & 0 & -s & c & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \\ U_6 \end{Bmatrix} \qquad (2.21b)$$

dengan

$c = \cos\alpha = \dfrac{X_j - X_i}{L}$, $s = \sin\alpha = \dfrac{Y_j - Y_i}{L}$ dan X_j = absis ujung j, X_i = absis ujung-i, Y_j = ordinat ujung-j serta Y_i = ordinat ujung-i.



Gambar 2.5. Transformasi koordinat

2.3.3. Hubungan antara gaya dalam sumbu lokal dan global

Dengan cara yang sama dapat diperoleh hubungan antara gaya batang {S} dalam koordinat lokal dengan gaya luar {P} dalam koordinat global, yaitu:

$$\{S\} = [T]\{P\} \qquad (2.22a)$$

$$\begin{Bmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \\ S_5 \\ S_6 \end{Bmatrix} = \begin{bmatrix} c & s & 0 & 0 & 0 & 0 \\ -s & c & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & c & s & 0 \\ 0 & 0 & 0 & -s & c & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \end{Bmatrix} \qquad (2.22b)$$

Karena $[T]$ adalah matriks bujur sangkar, maka pers. (2.10) dapat ditulis dalam bentuk lain sebagai

$$\{P\} = [T]^{-1}\{S\} \qquad (2.23)$$

Karena matriks transformasi $[T]$ adalah matriks ortogonal di mana $[T]^{-1} = [T]^T$, maka pers. (2.11) dapat ditulis menjadi

$$\{P\} = [T]^T\{S\} \qquad (2.24)$$

2.3.4. Matriks kekakuan dalam koordinat global

Dengan mensubstitusikan pers. (2.20), $\{S\} = [k]\{u\}$, pada pers. (24), $\{P\} = [T]^T\{S\}$, diperoleh

$$\{P\} = [T]^T\{S\}$$

$$\{P\} = [T]^T[k]\{u\}$$

Selanjutnya dengan mensubstitusikan pers. (2.21), $\{u\}=[T]\{U\}$, diperoleh

$$\{P\}=[T]^T[k][T]\{U\} \tag{2.25}$$

Pers. (2.25) merupakan hubungan antara gaya dalam koordinat global $\{P\}$ dan perpindahan dalam koordinat global $\{U\}$. Pers. (2.25) dapat disederhanakan menjadi

$$\{P\}=[K]\{U\} \tag{2.26}$$

dengan

$$[K]=[T]^T[k][T] \tag{2.27a}$$

$$[K]=\begin{bmatrix}
\left(\frac{EA}{L}c^2+\frac{12EI}{L^3}s^2\right) & \left(\frac{EA}{L}cs-\frac{12EI}{L^3}cs\right) & \left(-\frac{6EI}{L^2}s\right) & \left(-\frac{EA}{L}c^2-\frac{12EI}{L^3}s^2\right) & \left(-\frac{EA}{L}cs+\frac{12EI}{L^3}cs\right) & \left(-\frac{6EI}{L^2}s\right) \\
 & \left(\frac{EA}{L}s^2+\frac{12EI}{L^3}c^2\right) & \left(\frac{6EI}{L^2}c\right) & \left(-\frac{EA}{L}cs+\frac{12EI}{L^3}cs\right) & \left(-\frac{EA}{L}s^2-\frac{12EI}{L^3}c^2\right) & \left(\frac{6EI}{L^2}c\right) \\
 & & \frac{4EI}{L} & \left(\frac{6EI}{L^2}s\right) & \left(-\frac{6EI}{L^2}c\right) & \frac{2EI}{L} \\
 & & & \left(\frac{EA}{L}c^2+\frac{12EI}{L^3}s^2\right) & \left(\frac{EA}{L}cs-\frac{12EI}{L^3}cs\right) & \left(\frac{6EI}{L^2}s\right) \\
 & & & & \left(\frac{EA}{L}s^2+\frac{12EI}{L^3}c^2\right) & \left(-\frac{6EI}{L^2}c\right) \\
\text{simetri} & & & & & \frac{4EI}{L}
\end{bmatrix}$$

(27b)

2.3.5. Perakitan matriks kekakuan

Perakitan matriks kekakuan masing-masing batang, yang telah ditransformasikan dalam koordinat global, dilakukan dengan menggunakan vektor tujuan (Arfiadi, 2011, Balfour, 1986). Vektor tujuan merupakan nomor derajat kebebasan yang berkaitan dengan batang tersebut. Dengan cara ini maka proses perakitan dapat dilakukan dengan mudah, sehingga akhirnya diperoleh matriks kekakuan struktur, yaitu [K], sebagai:

$$[K] = \sum_{i=1}^{m} [K]_i \tag{2.28}$$

m = jumlah batang dalam struktur.

2.3.6. Vektor beban luar

Vektor beban luar selanjutnya ditentukan sesuai dengan urutan derajat kebebasan yang dipunyai struktur tersebut. Dengan ditetapkannya vector beban luar {P}, selanjutnya derajat kebebasan atau perpindahan global struktur dapat dihitung.

Untuk batang dengan beban pada bentangan, vektor beban luar terdiri dari beban yang bekerja pada titik simpul secara langsung dan beban titik ekivalen. Beban titik ekivalen diperoleh dari gaya jepit ujung yang dikerjakan dengan arah berlawanan dan dikerjakan pada koordinat global struktur. Arah yang berlawanan berarti secara matematik tanda gaya berlawanan dari tanda pada gaya jepit ujung batang. Karena gaya jepit ujung batang bekerja dalam arah koordinat batang, maka gaya harus ditransformasikan pada sumbu global. Hal ini dapat dicapai dengan mengalikan (pra-kali) gaya jepit dengan transpos dari matriks transformasi batang.

2.3.7. Perpindahan global

Vektor perpindahan struktur dapat diperoleh dari persamaan:

$$\{P\} = [K]\{U\} \tag{2.29}$$

Pers. (2.29) dapat diselesaikan dengan salah satu metode penyelesaian persamaan linier.

2.3.8. Perpindahan dalam koordinat lokal (deformasi batang)

Perpindahan batang dalam koordinat lokal (deformasi batang) selanjutnya dapat dihitung dari hubungan antara perpindahan sumbu lokal dan global menurut:

$$\{u\}_m = [T]_m \{U\}_m \tag{2.30}$$

Perlu dicatat bahwa $\{U\}_m$ pada pers. (18) merupakan bagian perpindahan global yang sesuai dengan batang yang ditinjau. Untuk struktur portal bidang $\{U\}_m$ berukuran 6 x 1, dan disusun sesuai dengan vektor tujuan batang tersebut.

2.3.9. Gaya batang

Gaya batang selanjutnya dapat dihitung dengan persamaan:

$$\{S\}_m = [k]_m \{u\}_m + \{S_o\}_m \tag{2.31}$$

dengan $\{S_o\}_m$ = gaya jepit ujung batang-m.

2.3.10. Pengaruh deformasi geser

Pada kasus-kasus tertentu, misal pada balok tinggi, selain pengaruh deformasi akibat lenturan, pengaruh deformasi geser dapat cukup besar nilainya. Apabila pengaruh deformasi geser akan diperhitungkan maka dapat diambil langkah sebagai berikut ini. Ditinjau suatu balok AB dengan bentangan L tanpa beban pada bentangannya. Persamaan *slope deflection* dapat ditulis sebagai

$$M_A = \frac{2EI}{L}\left(2\theta_A + \theta_B - 3\frac{\delta}{L}\right) \tag{2.32a}$$

$$M_B = \frac{2EI}{L}\left(\theta_A + 2\theta_B - 3\frac{\delta}{L}\right) \qquad (2.32b)$$

Perpindahan vertikal Δ terdiri dari perpindahan akibat deformasi lentur dan deformasi geser sebagai berikut:

$$\delta = \Delta + \Delta_s \qquad (2.33)$$

dengan Δ = pengaruh deformasi lentur dan Δ_s = pengaruh deformasi geser. Pengaruh deformasi geser dapat diperoleh dari persamaan:

$$\Delta_s = \frac{f_s VL}{GA} \qquad (2.34)$$

dengan f_s = faktor bentuk untuk geser, V = gaya geser dan G = modulus geser. Nilai f_s dihitung sebagai berikut:

$$f_s = \frac{A}{\overline{A}} \qquad (2.35)$$

dengan $\overline{A}$ = luas bidang geser efektif. Nilai f_s untuk beberapa bentuk tampang disajikan pada Tabel 2.1.

Tabel 2.1. Nilai faktor bentuk untuk geser f_s

Tampang	f_s
Empat persegi panjang	6/5
Lingkaran (masif)	10/9

Besarnya gaya geser dapat dinyatakan dengan

$$V = \frac{M_A + M_B}{L} \qquad (2.36a)$$

dan dengan mensubstitusikan pers. (2.32) sampai dengan (2.34) diperoleh:

$$V = \frac{6EI}{L^2(1+\alpha)}\left(\theta_A + \theta_B - 2\frac{\Delta}{L}\right) \tag{2.36b}$$

dengan

$$\alpha = \frac{12f_s EI}{GAL^2} \tag{2.37}$$

α merupakan koefisien deformasi geser.

Selanjutnya substitusikan pers. (2.36b) pada pers. (2.32a) dan (2.32b) sehingga diperoleh:

$$M_A = \frac{EI}{L}\left\{\left(\frac{4+\alpha}{1+\alpha}\right)\theta_A + \left(\frac{2-\alpha}{1+\alpha}\right)\theta_b - \left(\frac{6}{1+\alpha}\right)\frac{\Delta}{L}\right\} \tag{2.38a}$$

$$M_B = \frac{EI}{L}\left\{\left(\frac{2-\alpha}{1+\alpha}\right)\theta_A + \left(\frac{4+\alpha}{1+\alpha}\right)\theta_b - \left(\frac{6}{1+\alpha}\right)\frac{\Delta}{L}\right\} \tag{2.38b}$$

Apabila pers. (2.36b) dan pers. (2.38a)-(2.38b) serta pengaruh gaya aksial pada balok ditinjau, hubungan antara gaya internal dan deformasi dapat ditulis menurut persamaan:

$$\begin{Bmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \\ S_5 \\ S_6 \end{Bmatrix} = \begin{bmatrix} \frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\ 0 & \frac{12EI}{(1+\alpha)L^3} & \frac{6EI}{(1+\alpha)L^2} & 0 & -\frac{12EI}{(1+\alpha)L^3} & \frac{6EI}{(1+\alpha)L^2} \\ 0 & \frac{6EI}{(1+\alpha)L^2} & \frac{(4+\alpha)EI}{(1+\alpha)L} & 0 & -\frac{6EI}{(1+\alpha)L^2} & \frac{(2-\alpha)EI}{(1+\alpha)L} \\ -\frac{EA}{L} & 0 & 0 & \frac{EA}{L} & 0 & 0 \\ 0 & -\frac{12EI}{(1+\alpha)L^3} & -\frac{6EI}{(1+\alpha)L^2} & 0 & \frac{12EI}{(1+\alpha)L^3} & -\frac{6EI}{(1+\alpha)L^2} \\ 0 & \frac{6EI}{(1+\alpha)L^2} & \frac{(2-\alpha)EI}{(1+\alpha)L} & 0 & -\frac{6EI}{(1+\alpha)L^2} & \frac{(4+\alpha)EI}{(1+\alpha)L} \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{Bmatrix} \tag{2.39a}$$

atau

$$\{S\} = [k]\{u\} \tag{2.39b}$$

Jika balok mempunyai beban pada bentangan, maka gaya internal dihitung dengan

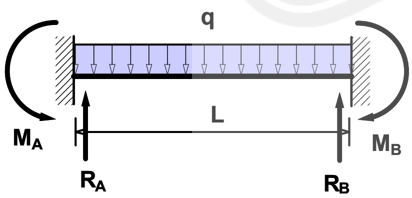
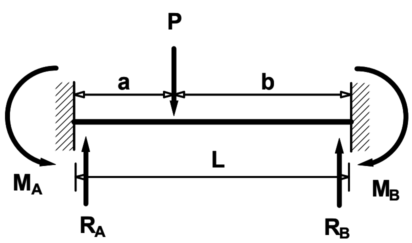
$$\{S\}=[k]\{u\}+\{S_o\} \tag{2.39c}$$

Dalam hal ini

$$[k]=\begin{bmatrix} \frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\ 0 & \frac{12EI}{(1+\alpha)L^3} & \frac{6EI}{(1+\alpha)L^2} & 0 & -\frac{12EI}{(1+\alpha)L^3} & \frac{6EI}{(1+\alpha)L^2} \\ 0 & \frac{6EI}{(1+\alpha)L^2} & \frac{(4+\alpha)EI}{(1+\alpha)L} & 0 & -\frac{6EI}{(1+\alpha)L^2} & \frac{(2-\alpha)EI}{(1+\alpha)L} \\ -\frac{EA}{L} & 0 & 0 & \frac{EA}{L} & 0 & 0 \\ 0 & -\frac{12EI}{(1+\alpha)L^3} & -\frac{6EI}{(1+\alpha)L^2} & 0 & \frac{12EI}{(1+\alpha)L^3} & -\frac{6EI}{(1+\alpha)L^2} \\ 0 & \frac{6EI}{(1+\alpha)L^2} & \frac{(2-\alpha)EI}{(1+\alpha)L} & 0 & -\frac{6EI}{(1+\alpha)L^2} & \frac{(4+\alpha)EI}{(1+\alpha)L} \end{bmatrix} \tag{2.40}$$

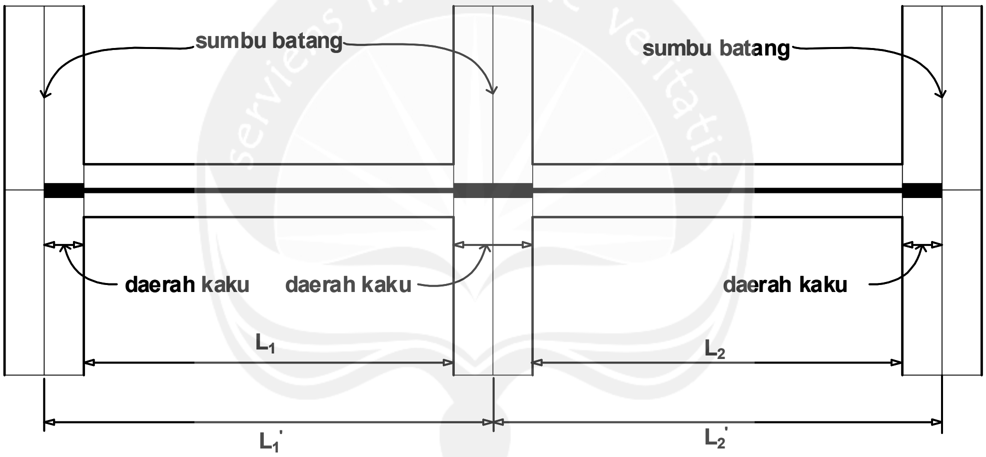
Momen jepit dapat diturunkan dengan metode klasik seperti metode luasan momen, beban satuan atau metode klasik lainnya. Beberapa nilai momen jepit disajikan dalam Tabel 2.2.

Tabel 2.2. Gaya jepit ujung batang dengan memperhitungkan deformasi geser

No	Kondisi	GAYA JEPIT UJUNG KIRI	GAYA JEPIT UJUNG KANAN
1		$M_A=\frac{1}{12}qL^2$ $R_A=\frac{1}{2}qL$	$M_B=-\frac{1}{12}qL^2$ $R_B=\frac{1}{2}qL$
2		$M_A=\frac{Pab^2}{L^2}\left\{\frac{1+\frac{6\beta}{bL}}{1+\frac{12\beta}{L^2}}\right\}$ $\beta=\frac{f_sEI}{GA}$ $R_A=\frac{1}{L}\{abs(M_A)-abs(M_B)+Pb\}$	$M_A=-\frac{Pa^2b}{L^2}\left\{\frac{1+\frac{6\beta}{aL}}{1+\frac{12\beta}{L^2}}\right\}$ $\beta=\frac{f_sEI}{GA}$ $R_A=\frac{1}{L}\{abs(M_B)-abs(M_A)+Pa\}$

2.3.11. Daerah kaku pada pertemuan balok-kolom

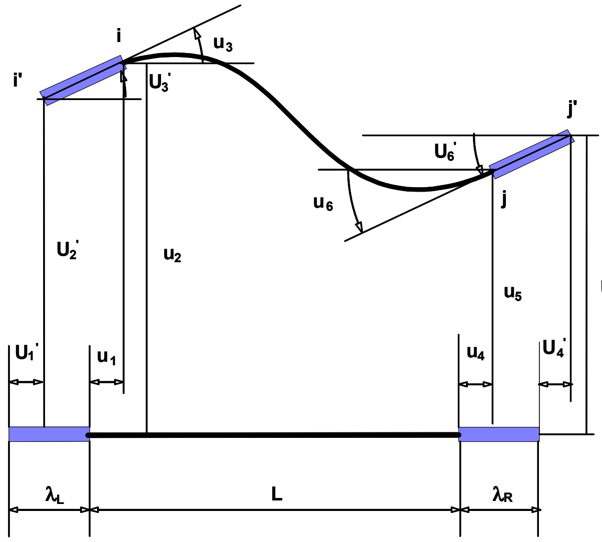
Dalam tinjauan analisis struktur rangka, batang-batang dianggap sebagai elemen garis. Dalam hal ini titik kumpul merupakan perpotongan dari sumbu-sumbu batang, sehingga panjang bentangan diukur dari pusat ke pusat. Dalam kenyataan sesungguhnya, pada ujung-ujung batang terdapat daerah yang sangat kaku, yaitu dari pusat pertemuan batang-batan (titik kumpul) ke sisi-sisi (bidang muka) batang seperti terlihat pada Gambar 5.20. Dalam perencanaan suatu struktur, misal balok beton bertulang, gaya-gaya dalam dapat diambil sebagai gaya-gaya pada bidang muka kolom sehingga hasil perencanaan dapat lebih ekonomis.



Gambar 2.6. Daerah ujung kaku balok

Untuk batang dengan daerah kaku pada ujung-ujung batang, matriks kekakuan batang pertama-tama diturunkan untuk bagian dalam (tanpa daerah kaku). Oleh karena itu perlu dibentuk matriks transformasi dari deformasi batang tanpa daerah kaku ke deformasi batang pada ujung-ujung.

Hubungan antara deformasi pada ujung balok bagian-dalam dan ujung balok bagian-luar dapat ditentukan dengan memperhatikan Gambar 5.21 sebagai berikut:



Gambar 2.7. Hubungan antara deformasi pada batang dengan daerah kaku pada ujung

$$u_1 \approx U_1^{'} \tag{2.41a}$$

$$u_2 = U_2^{'} + \lambda_L U_3^{'} \tag{2.41b}$$

$$u_3 = U_3^{'} \tag{2.41c}$$

$$u_4 \approx U_4^{'} \tag{2.41d}$$

$$u_5 = U_5^{'} - \lambda_R U_6^{'} \tag{2.41e}$$

$$u_6 = U_6^{'} \tag{2.41f}$$

Pers. (2.41a)-(2.41f) dapat ditulis dalam bentuk matriks:

$$\begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{Bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & \lambda_L & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -\lambda_R \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} U_1^{'} \\ U_2^{'} \\ U_3^{'} \\ U_4^{'} \\ U_5^{'} \\ U_6^{'} \end{Bmatrix} \tag{2.42}$$

atau secara lebih sederhana:

$$\{u\} = [T^{'}]\{U^{'}\} \tag{2.43}$$

dengan matriks transformasi $\left[T'\right]$:

$$\left[T'\right]=\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & \lambda_L & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -\lambda_R \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.44}$$

Apabila matriks kekakuan batang yang berkaitan dengan deformasi ujung dalam balok dinyatakan dengan $[k]$ menurut pers. (2.20b) yaitu

$$[\mathbf{k}]=\begin{bmatrix} \frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\ 0 & \frac{12EI}{L^3} & \frac{6EI}{L^2} & 0 & -\frac{12EI}{L^3} & \frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{4EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{2EI}{L} \\ -\frac{EA}{L} & 0 & 0 & \frac{EA}{L} & 0 & 0 \\ 0 & -\frac{12EI}{L^3} & -\frac{6EI}{L^2} & 0 & \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{2EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{4EI}{L} \end{bmatrix}$$

maka matriks kekakuan batang yang berkaitan dengan deformasi ujung luar batang $\left[k'\right]$ dapat diperoleh dengan persamaan:

$$\left[k'\right]=\left[T'\right]^T[k]\left[T'\right] \tag{2.45}$$

di mana dengan mengingat pers. (2.44) dan pers. (2.20b) diperoleh

$$
[k'] = \begin{bmatrix}
\frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\
 & \frac{12EI}{L^3} & \left(\frac{12EI}{L^3}\lambda_L + \frac{6EI}{L^2}\right) & 0 & -\frac{12EI}{L^3} & \left(\frac{12EI}{L^3}\lambda_R + \frac{6EI}{L^2}\right) \\
 & & \left(\frac{12EI}{L^3}\lambda_L^2 + \frac{12EI}{L^2}\lambda_L + \frac{4EI}{L}\right) & 0 & \left(-\frac{12EI}{L^3}\lambda_L - \frac{6EI}{L^2}\right) & \left(\frac{12EI}{L^3}\lambda_L\lambda_R + \frac{6EI}{L^2}\lambda_R + \frac{6EI}{L^2}\lambda_L + \frac{2EI}{L}\right) \\
 & & & \frac{EA}{L} & 0 & 0 \\
 & & & & \frac{12EI}{L^3} & \left(-\frac{12EI}{L^3}\lambda_R - \frac{6EI}{L^2}\right) \\
 & & & & & \left(\frac{12EI}{L^3}\lambda_R^2 + \frac{12EI}{L^2}\lambda_R + \frac{4EI}{L}\right)
\end{bmatrix}
$$

(2.46)

Untuk batang dengan deformasi geser matriks kekakuan $[k']$ dapat diturunkan secara sama, yaitu dengan mengganti $[k]$ pada pers. (2.45) dengan pers. (2.40).

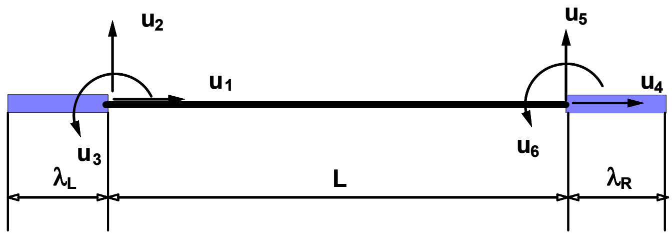
Selanjutnya apabila derajat kebebasan diukur pada pertemuan batang (bagian luar ujung balok dengan daerah kaku), matriks kekakuan dalam koordinat global dapat ditentukan dengan cara mentransformasikan matriks kekakuan $[k']$ ke dalam koordinat global menurut:

$$[K] = [T]^T [k'][T] \tag{2.47}$$

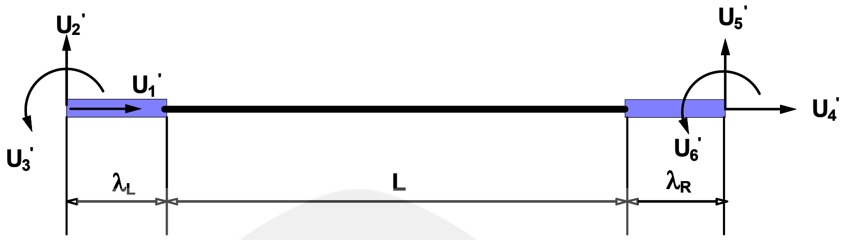
Atau dengan mengingat pers. (2.45) diperoleh:

$$[K] = [T]^T [T']^T [k][T'][T] \tag{2.48}$$

Persamaan (2.48) berlaku baik untuk batang yang memperhitungkan deformasi geser atau tidak dengan memasukkan nilai matriks kekakuan batang $[k]$ yang sesuai. Langkah hitungan selanjutnya sama seperti langkah hitungan pada pasal sebelumnya dengan mengacu pada derajat kebebasan bagian luar balok seperti ditunjukkan pada Gambar 2.8



a. Derajat kebebasan pada bagian dalam balok



b. Derajat kebebasan pada bagian luar balok

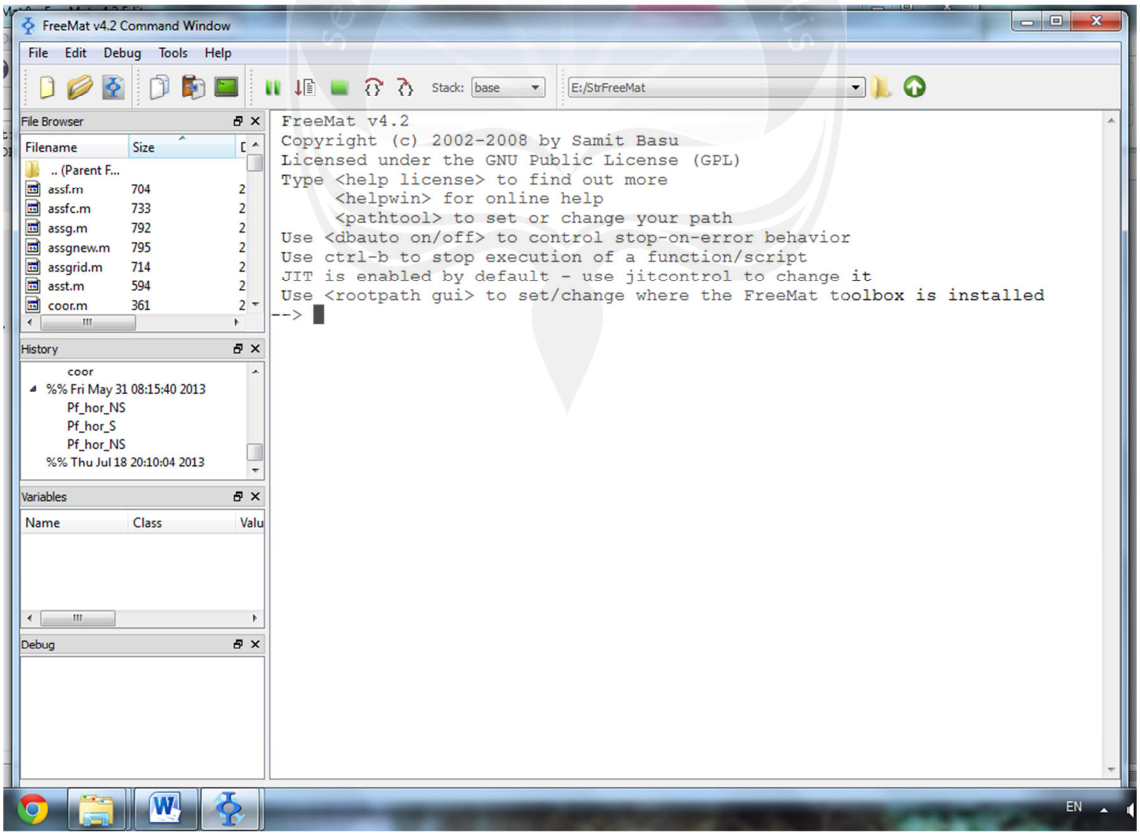
Gambar 2.8. Derajat kebebasan batang dengan daerah kaku

BAB III
PENGEMBANGAN PROGRAM

3.1. PENDAHULUAN

Program Freemat adalah program *open source* seperti Matlab, yang bermanfaat untuk penyelesaian pemrograman numerik dalam bidang sains dan teknik. Freemat kira-kira 95% kompatibel dengan Matlab. Karena Freemat merupakan *open source* program, maka sangat cocok untuk digunakan di kalangan perguruan tinggi karena tidak perlu untuk membayar lisensi yang biasanya relatif mahal. Freemat dapat diunduh di **freemat**.sourceforge.net/.

Setelah Freemat ada dalam komputer, jika dijalankan akan muncul jendela seperti pada Gambar 3.1.



Gambar 3.1. Jendela Freemat

Program yang dikembangkan dalam REALIN disimpan sebagai berkas dengan ekstensi m. Subprogram ini dapat dipanggil oleh program atau perintah lain dalam Freemat.

Dengan pendekatan masukan dan keluaran, subprogram-subprogram dikembangkan sebagai alat untuk melakukan analisis struktur dengan metode matriks kekakuan. Subprogram yang dikembangkan harus sedemikian sehingga pengguna dapat memahami mengenai kebutuhan masukan dan keluaran yang dihasilkan (Gambar 3.2). Masukan tergantung pada keluaran yang diinginkan dalam setia subprogram. Sedangkan proses hitungan dilakukan oleh program. Langkah hitungan secara keseluruhan dirakit dari gabungan-gabungan subprogram dan ditentukan oleh pengguna. Dengan demikian program secara keseluruhan sangat berguna untuk alat pembelajaran metode matriks kekakuan.

Gambar 3.2. Masukan, proses dan keluaran

Dalam pengembangan program, terdapat tiga macam program sesuai kegunaannya, yaitu: program yang berlaku umum, program khusus untuk rangka bidang, dan program khusus untuk portal bidang.

3.2. PROGRAM YANG BERLAKU UMUM

3.2.1. Program coor.m

Dalam analisis struktur suatu titik dapat dutetapkan dengan suatu koordinat dalam suatu salib sumbu Cartesian. Letak salib sumbu Cartesian dapat dipilih bebas. Dengan diketahuinya letak suatu titik dalam salib sumbu Cartesian, maka panjang batang yang menghubungkan dua titik dapat dihitung oleh program. Selain itu orientasi atau arah batang juga dapat diketahui. Program coor.m merupakan implementasi untuk menentukan letak suatu titik pada salib sumbu Cartesian sebagai berikut ini.

```
function n=coor(x,y)
% COOR.M produces coordinate (x,y) for a specified node
% N1=coor(x1,y1)
% N1 = node number 1
% x1,y1 = coordinate of node number 1

% developed Yoyong Arfiadi
% Atma Jaya Yogyakarta University
% 9610--ver. 1.0
% 9703--ver 1.1
% 130718 v1.2
n=[x, y];
```

3.2.2. Program kg.m

Pogram kg.m digunakan untuk mentransformasikan matriks kekakuan dalam koordinat lokal ke dalam matriks kekakuan dalam koordinat global. Program kg.m adalah sebagai berikut ini.

```
function K=kg(k,T)
% KG generate stiffness matrix in global coordinate
% kg(k,T)
% where k = stiffness matrix in local coordinate
% T = transformation coordinate in function
% memt for the plane truss structures
% memf for the plane frame structures

% Yoyong Arfiadi
% Atma Jaya Yogyakarta University
% 960903 1.0
% 130718 v1.1
%T=[c s 0 0;-s c 0 0;0 0 c s; 0 0 -s c];
K=T'*k*T;
```

Sebagai input pada program kg.m adalah matriks kekakuan dalam koordinat lokal k dan matriks transformasi T, sedangkan keluarannya adalah matriks kekakuan dalam koordinat global K.

3.2.3. Program solv.m

Program solv.m berfungsi untuk menyelesaikan persamaan linier simultan. Program solv.m sebagai berikut ini.

```
function U=solv(K,P)
% SOLV solves simultaneous equation
% K U = P
% U = SOLV(K,P)

%developed Yoyong Arfiadi
%9611-- 1.0
%130718 v1.1
%Atma Jaya Yogyakarta University

U=K\P;
```

3.2.4. Program plot2d.m

Program plot2d.m berfungsi untuk menggambar struktur 2 dimensi dengan data hubungan antar titik simpul. Program plot2d.m sebagai berikut ini.

```
function figure = plot2d(N,linestyle,linewidth,t)
%plot2d.m      draw mesh 2D structure
%              N = mesh of structure containing connecting node
%              linestyle = line style
%              linewidth = line width == t

% developed Yoyong Arfiadi
% Atma Jaya Yogyakarta University
% 130724 v0.0

size(N,1);
hold on
jmlhnode=2;
for ii=1:jmlhnode:size(N,1)
   hold on
    plot( N(ii:ii+1,1),N(ii:ii+1,2),linestyle,linewidth,t)
    axis equal
end

axis off
```

3.3. PROGRAM UNTUK RANGKA BIDANG

3.3.1. Program memt.m

Program memt.m digunakan untuk menghitung panjang batang dan matriks transformasi batang dari data koordinat titik simpul.

Jika koordinat titik simpul diketahui, maka panjang batang L yang menghubungkan dua titik dapat dihitung dengan persamaan:

$$L = \sqrt{\left(X_j - X_i\right)^2 + \left(Y_j - Y_i\right)^2} \tag{3.1}$$

dengan X_j = absis titik –j, X_i = absis titik –i, Y_j = ordinat titik –j, Y_i = ordinat titik –i.

Program memt.m adalah sebagai berikut ini.

```
function [l,T] = memt(ci,cj)
% MEMT generates member length and transformation matrix
% for the plane truss structures
% [L,T] = MEMT(CI,CJ)
% L = length of the member
% T = matrix transformation
% CI = I node number of the element
% CJ = J node number of the element
% NOTE: coordinates of CI, CJ have been defined
% in COOR.M
% see also MEMF.M

% developed Yoyong Arfiadi
% Atma Jaya Yogyakarta University
% 961002 v1.0
% 130718 v1.1

l=sqrt(sum((ci-cj).^2));
%l = sqrt((cj(1,1) - ci(1,1))^2 + (cj(1,2) - ci(1,2))^2));
c=(cj(1,1)-ci(1,1))/l;
s=(cj(1,2)-ci(1,2))/l;
T=[c s 0 0;-s c 0 0;0 0 c s;0 0 -s c];
```

3.3.2. Program klt.m

Program klt.m berfungsi untuk membentuk matriks kekakuan batang dalam koordinat lokal untuk struktur rangka bidang. Program klt.m adalah sebagai berikut ini.

```
function k=klt(E,A,L)
% KLT generates member stiffness matrix for truss structures
% in local coordinate
% k=KLT(E,A,L)
% E = modulus of elasticity,
% A = area of the element,
% L = length of the member

% developed Yoyong Arfiadi
% Atma Jaya Yogyakarta University
% 960903 v1.0
% 9612-- v1.01
% 130718 v1.1
k1=A*E/L;
k=[k1 0 -k1 0;0 0 0 0;-k1 0 k1 0; 0 0 0 0];
```

3.3.3. Program asst.m

Program asst.m berfungsi untuk menggabungkan matriks kekakuan batang dalam koordinat global menjadi matriks kekakuan struktur. Penggabungan dilakukan berdasarkan vektor tujuan ID masing-masing batang. Program asst.m adalah sebagai berikut ini.

```
function ka=asst(kgt,id,tdf)
% KA = ASST(KGT,ID,TDF)
%     assembly stiffness matrix KGT according to ID vector and
%     Total degree of freedom of the structure
%           KA  = the contributon of KGT to the structure
stiffness matrix
%           KGT = stiffness matrix to be assembled
%           ID  = ID vector
%           TDF = total degree of freedom of the structure

%     developed Yoyong Arfiadi
%     Atma Jaya Yogyakarta University
%   130719  v1.1

ka=zeros(tdf,tdf);
idt=id';
for i=1:4
  ii=id(1,i);
    for j = 1:4
      ij=idt(j,1);
      if (ii~=0 & ij ~=0)
        ka(ii,ij)=kgt(i,j);
      end
    end
end
```

3.3.4. Program disst.m

Program disst.m merupakan program untuk menghitung perpindahan dalam koordinat lokal (deformasi) masing-masing batang. Program disst.m adalah sebagai berikut ini.

```
function u=disst(U,id,T)
% DISST         calculates local degree of freedom
%               for plane truss structures
%               u = DISST(U,id,T)
%                       u  = member deformation (local)
%                       U  = structure displacement (global)
%                       id = vector destination of the member
%                       T  = transformation matrix
% see also: DISSF.M

% developed Yoyong Arfiadi
% 9611-- 1.0
% 130719   v1.1
% 0512-- == 'dist' is  in ANN toolbox --'disst'
% Atma Jaya Yogyakarta University

for i=1:4
   ii =id(1,i);
   if (ii == 0);
      Ul(i,1) = 0;
   else
      Ul(i,1)=U(ii,1);
  end;
end;
u=T*Ul;
```

3.3.5. Program stret.m

Program stret.m berfungsi untuk menghitung gaya batang. Program stret.m adalah sebagai berikut ini.

```
function S=stret(k,u)
% STRET(k,u) produces internal forces for plane truss structures
% S=STRET(k,u)
% S=internal forces
% k=member stiffness matrix in local coordinate
% u=member deformation vector

% developed Yoyong Arfiadi
% 961004 1.0
% 130719  v1.0
S=k*u;
```

3.3.6. Rekapitulasi program untuk rangka bidang

Seluruh program yang digunakan dalam analisis rangka bidang dapat dilihat pada Tabel 3.1.

Tabel 3.1. Subprogram untuk rangka bidang

NO	PROGRAM	FUNGSI	*OUTPUT*	*INPUT*	*SYNTAX*
1	coor.m	Menghasilkan koordinat titik nodal	Ni(Xi,Yi)	Xi, Yi	Ni=coor(Xi,Yi)
2	memt.m	Menghitung panjang batang dan matriks transformasi batang	Lm, Tm	Ni, Nj	[Lm,Tm]=memt(Ni,Nj)
3	klt.m	Membentuk matriks kekakuan batang untuk sistem plane truss dalam koordinat lokal	km	Em, Am,Lm	km=klt(Em,Am,Lm)
4	kg.m	Melakukan transformasi matriks kekakuan batang dari koordinat lokal ke koordinat global	Km	km, Tm	Km=kg(km,Tm)
5	asst.m	Menggabungkan matriks kekakuan batang dalam koordinat global ke dalam matriks kekakuan struktur	Kmi	Km, IDm, TDF	Kmi=asst(Km,IDm,TDF)
6	solv.m	Menyelesaikan persamaan linier simultan	U	K, P	U=solv(K,P)
7	disst.m	Menghitung deformasi batang	um	U, IDm,Tm	um=disst(U,IDm,Tm)
8	stret.m	Menghitung gaya batang untuk struktur *plane truss*	Sm	km,um	Sm=stret(km,um)
9	plot2d.m	Menggambar struktur 2D	gambar	N, linestyle,linewidth	figure = plot2d(N,linestyle,linewidth,t)

Notasi pada Tabel 3.1 adalah sebagai berikut ini.

Ni = titik simpul i

Xi = absis titil simpul i

Yi = ordinat titik simpul i

Lm = panjang batang m

Tm = matriks transformasi batang m

Em = modulus elastisitas batang m

Am = luas tampang batang m

km = matriks kekakuan batang m dalam koordinat lokal

Km = matriks kekakuan batang m dalam koordinat global

IDm = vektor tujuan batang m

TDF = jumlah derajat kebebasan struktur

Kmi = kontribusi matriks kekakuan batang m pada matriks kekakuan struktur

K = matriks kekakuan struktur

P = vektor beban luar

U = perpindahan global struktur (derajat kebebasan)

um = perpindahan (deformasi) batang m

Sm = gaya internal batang m

3.4. PROGRAM UNTUK PORTAL BIDANG

3.4.1. Program klf.m

Program klf.m berfungsi membentuk matriks kekakuan lokal suatu batang terlentur tanpa deformasi geser (balok Bernouli). Jika deformasi geser diperhitungkan (balok Timoshenko) maka dapat digunakan program klfs.m. Program klf.m adalah sebagai berikut ini.

```
function k=klf(E,A,I,L)
% klf generates member stiffness matrix
% for plane frame in local coordinate
% k = klf(E,A,I,L)
% k = member stiffness matrix
% E = modulus of elasticity
% A = area
% I = second moment area
% L = length of the member
% see also: KLFS.M, KLFM.M, KLT.M

% developed Yoyong Arfiadi
% Atma Jaya Yogyakarta University
% 960925 1.0
% 130719 v1.1

k1=12;
k2=6;
k3=6;
k4=4;
```

```
k5=-6;
k6=2;
k7=-6;
k8=4;
a0=E*A/L;
a1=E*I/L;
a2=a1/L;
a3=a2/L;
k=[a0 0 0 -a0 0 0;
0 k1*a3 k2*a2 0 -k1*a3 k3*a2;
0 k2*a2 k4*a1 0 k5*a2 k6*a1;
-a0 0 0 a0 0 0;
0 -k1*a3 k5*a2 0 k1*a3 k7*a2;
0 k3*a2 k6*a1 0 k7*a2 k8*a1];
```

3.4.2. Program klfs.m

Program klfs.m berfungsi untuk membentuk matriks kekakuan batang terlentur dengan deformasi geser (balok Timoshenko). Program klfs.m adalah sebagai berikut ini.

```
function k=klfs(E,A,I,L,f,v)
% klfs generates member stiffness matrix with shear deformation
% for plane frame in local coordinate
% k = KLF(E,A,I,L,f,v)
% k = member stiffness matrix
% E = modulus of elasticity
% A = area
% I = second moment area
% L = length of the member
% f = shear area reduction
% f = A/Abar
% Abar = effective shear area
% v = Poisson's ratio
% see also: KLF.M, KLFM.M, KLT.M

% developed Yoyong Arfiadi
% Atma Jaya Yogyakarta University
% 050902 1.0
% 130719 v1.01

G=E/(2*(1+v));

a0=E*A/L;

a=12*f*E*I/(G*A*L^2);

k=[a0 0 0 -a0 0 0;
0 12*E*I/((1+a)*L^3) 6*E*I/((1+a)*L^2) 0 -12*E*I/((1+a)*L^3) 6*E*I/((1+a)*L^2);
```

```
0 6*E*I/((1+a)*L^2) (4+a)*E*I/((1+a)*L) 0 -6*E*I/((1+a)*L^2) (2-
a)*E*I/((1+a)*L)
-a0 0 0 a0 0 0;
0 -12*E*I/((1+a)*L^3) -6*E*I/((1+a)*L^2) 0 12*E*I/((1+a)*L^3) -
6*E*I/((1+a)*L^2);
0 6*E*I/((1+a)*L^2) (2-a)*E*I/((1+a)*L) 0 -6*E*I/((1+a)*L^2)
(4+a)*E*I/((1+a)*L)];
```

3.4.3. Program klfm.m

Program klfm.m berfungsi untuk membentuk matriks kekakuan batang dengan ujung termodifikasi. Program klfm,m adalah sebagai berikut ini.

```
function k=klfm(E,A,I,L,c)
% KLFM      generates modified member stiffness matrices
%      for plane frame structure in local coordinate
%      k = KLFM(E,A,I,L,c)
%            k = member stiffness matrix
%            E = modulus of elasticity
%            A = area
%            I = second moment area
%            L = length
%            c = 2   : fixed - pinned   |--------o|
%
%              = 3   : pinned - fixed   |o--------|
%
%              = 4   : pinned - pinned  |o-------o|
%
% see also: KLF.M, KLT.M, KLFS.M

% developed Yoyong Arfiadi
%   9611--
%   130719 v1.0
% Atma Jaya Yogyakarta University

if (c==2) % fixed - pinned
k1=3;
k2=3;
k3=0;
k4=3;
k5=-3;
k6=0;
k7=0;
k8=0;

elseif (c==3) %pinned - fixed
      k1=3;
      k2=0;
      k3=3;
```

```
        k4=0;
        k5=0;
        k6=0;
        k7=-3;
        k8=3;

elseif (c==4) % pinned - pinned
        k1=0;k2=0;k3=0;k4=0;k5=0;k6=0;k7=0;k8=0;
end;

        if(c~=2 & c~=3 & c~=4) error('Boundary Condition Fail')
        end;
a0=E*A/L;
a1=E*I/L;
a2=a1/L;
a3=a2/L;
k=[a0    0      0      -a0    0       0;
   0  k1*a3  k2*a2   0   -k1*a3  k3*a2;
   0  k2*a2  k4*a1   0    k5*a2  k6*a1;
  -a0   0      0      a0    0       0;
   0 -k1*a3   k5*a2  0    k1*a3  k7*a2;
   0  k3*a2  k6*a1    0   k7*a2  k8*a1];
```

3.4.4. Program assf.m

Program assf.m berfungsi untuk merakit matriks kekakuan batang dalam koordinat global ke dalam matriks kekakuan struktur sesuai vektor tujuan batang tersebut. Program assf.m adalah sebagai berikut ini.

```
function ka=assf(kgf,id,tdf)
% KA = ASSF(KGF,ID,TDF)
%     assembly stiffness matrix KGF according to ID vector and
%     Total degree of freedom of the structure,
%     with the possibility to have a constraint (the same id
number)
%     KA  = the contributon of KGF to the structure stiffness
matrix
%     KGF = stiffness matrix to be assembled
%     ID  = ID vector
%     TDF = total degree of freedom of the structure

%     developed Yoyong Arfiadi
%     020403 v1.2
%     130719
%     Atma Jaya Yogyakarta University
```

```
ka=zeros(tdf,tdf);
idt=id';
for i=1:6
  ii=id(1,i);
    for j = 1:6
      ij=idt(j,1);
      if (ii~=0 & ij ~=0)
        ka(ii,ij)=ka(ii,ij)+kgf(i,j);
      end
    end
end
```

3.4.5. Program feq.m

Program ini berfungsi untuk menghitung gaya jepit oleh beban tebagi rata sebagian.

Program feq.m sebagai berikut ini.

```
function so=feq(q,a1,a2,L)
% FEQ   produces fixed end forces due to uniform load
%  so = feq(q,a1,a2,L)
%             so = fixed end forces
%             q  = uniform load intensity
%             a1 = distance from the left node
%                  to the left-end load
%             a2 = distance from the left node
%                  to the right-end load
%             L  = length of the member
% see also: FEP.M, FEG.M, FET.M, FEPM.M, FEQM.M, FETM.M

% developed Yoyong Arfiadi
% 9611-- 1.0
% 130720 v1.01
% Atma Jaya Yogyakarta University

d=a2-a1;
b=L-a1;
s1=0;
s3=-q/L^2*(a1*b^2*d+0.5*(b^2-2*a1*b)*d^2+1/3*(a1-
2*b)*d^3+1/4*d^4); %because qdown (-)
s4=0;
s6=q/L^2*(a1^2*b*d+1/2*(2*a1*b-a1^2)*d^2+1/3*(b-2*a1)*d^3-
1/4*d^4);
s2=-q*d*(L-a1-0.5*d)/L + 1/L*(s3+s6);
s5=-q*d-s2;
so=[s1;s2;s3;s4;s5;s6];
```

3.4.6. Program fep.m

Program fep.m berfungsi untuk menghitung gaya jepit ujung batang oleh beban terpusat, yang bekerja dalam jarak tertentu dari titik simpul pertama. Program fep.m adalah sebagai berikut ini.

```
function soa=fep(p1,x1,L)
% fep   produces fixed end forces due to point load
%       soa = fep(p1,x1,L)
%       soa = fixed end forces
%               p1  = point load
%               x1  = p1 distance from 1st node
%       L   = length of the member
% see also: FEP3.M, FEQ.M, FEG.M, FET.M, FEPM.M, FEQM.M, FETM.M

% developed Yoyong Arfiadi
% 9611-- 0.0
% 130720 v1.0
% Atma Jaya Yogyakarta University

s1=0;
s3=-(1/L^2*(p1*x1*(L-x1)^2 ));  %because qdown (-)
s4=0;
s6=1/L^2*(p1*x1^2*(L-x1));
s2=1/L*(-p1*(L-x1)) + (s3+s6)/L;
s5=1/L*(-p1*x1) - (s3+s6)/L;
soa=[s1;s2;s3;s4;s5;s6];
```

3.4.7. Program fep3.m

Program fep3.m menghitung gaya jepit ujung batang oleh tiga beban terpusat. Program fep3.m adalah sebagai berikut ini.

```
function soa=fep3(p1,x1,p2,x2,p3,x3,L)
% fep3  produces fixed end forces due to 3 point loads
%       soa = fep3(p1,x1,p2,x2,p3,x3,L)
%           soa = fixed end forces
%           pi  = i th point load
%           xi  = pi distance from 1st node
%           L   = length of the member
% see also: FEP.M, FEQ.M, FEG.M, FET.M, FEPM.M, FEQM.M, FETM.M

% developed Yoyong Arfiadi
% 9611-- 0.0
% 130720  v1.0
% Atma Jaya Yogyakarta University
```

```
s1=0;
s3=-(1/L^2*(p1*x1*(L-x1)^2 + p2*x2*(L-x2)^2 + p3*x3*(L-x3)^2));
s4=0;
s6=1/L^2*(p1*x1^2*(L-x1) + p2*x2^2*(L-x2) + p3*x3^2*(L-x3));
s2=((1/L*(-p1*(L-x1)-p2*(L-x2)-p3*(L-x3))) + (s3+s6)/L);
s5=((1/L*(-p1*x1-p2*x2-p3*x3)) - (s3+s6)/L);
soa=[s1;s2;s3;s4;s5;s6];
```

3.4.8. Program feg.m

Program feg.m berfungsi untuk menghitung gaya jepit ujung oleh beban segitiga. Program feg.m adalah sebagai berikut ini.

```
function so=feg(q,L)
% feg   produces fixed end forces due to triangular load
%       so = feg(q,L)
%           so = fixed end forces
%           q   = triangular load intensity
%           L   = length of the member
% see also: FEP.M, FEQ.M, FET.M,FEPM.M,FEQM.M,FETM.M

% developed Yoyong Arfiadi
% 9611-- 0.0
% 130730 v1.0
% Atma Jaya Yogyakarta University

s1=0;
s2=-0.25*q*L;
s3=-5/96*q*L^2;
s4=0;
s5=s2;
s6=-s3;
so=[s1;s2;s3;s4;s5;s6];
```

3.4.9. Program fet.m

Program fet.m berfungsi untuk menghitung gaya jepit ujung batang oleh beban trapesium. Program fet.m adalah sebagai berikut ini.

```
function so=fet(q,a,L)
% FET   produces fixed end forces due to
%       trapezoidal load
%       so = FET(q,a,L)
%           so = fixed end forces
```

```
%            q  = load intensity
%            a  = distance from first node
%                 to the first peak load
%            L  = length of the member
% see also: FEQ.M, FEP.M, FEG.M, FEPM.M, FEGM.M, FEQM.M, FETM.M

% developed Yoyong Arfiadi
% 9611-- 00
% 130720 v1.0
% Atma Jaya Yogyakarta University

if(2*a>L) error('Not a Trapezoidal Load');end;
s1=0;
s2=-(0.5*q*a + 0.5*q*(L-2*a));
s3=-1/12*q*L^2*(1-2*(a/L)^2 + (a/L)^3);
s4=0;
s5=s2;
s6=-s3;
so=[s1;s2;s3;s4;s5;s6];
```

3.4.10. Program feg2.m

Program feg2.m menghitung gaya jepit ujung batang oleh dua buah beban segitiga simetri. Program feg2.m adalah sebagai berikut ini.

```
function so=feg2(q,L)
% feg2  produces fixed end forces due to 2 symmetric
%       triangular load
%       so = feg2(q,L)
%               so = fixed end forces
%               q    = triangular load intensity
%               L    = length of the member
% see also: FEP.M, FEP.M, FEQ.M, FEPM.M, FEQM.M

% developed Yoyong Arfiadi
% 9611-- 0.0
% 130720 v1.0
% Atma Jaya Yogyakarta University

s1=0;
s2=-0.25*q*L;
s3=-17/384*q*L^2;
s4=0;
s5=s2;
s6=-s3;
so=[s1;s2;s3;s4;s5;s6];
```

3.4.11. Program feqm.m

Program feqm.m berfungsi untuk menghitung gaya jepit ujung batang termodifikasi oleh beban terbagi rata. Program feqm.m adalah sebagai berikut ini.

```
function so=feqm(q,a1,a2,L,c)
% feqm  produces fixed end forces due to uniform load
%       for modified member
%       so = feqm(q,a1,a2,L,c)
%       so = fixed end forces
%               q  = uniform load intensity
%               a1 = distance from the left node
%                    to the left-end load
%               a2 = distance from the left node
%                    to the right-end load
%               L  = length of the member
%               c  = 2      |---------o
%                    3     o---------|
% see also: FEP.M, FEG.M, FET.M, FEPM.M, FETM.M

% developed Yoyong Arfiadi
% 9611.. 0.0
% 130720 v1.0
% Atma Jaya Yogyakarta University

d=a2-a1;
a=a1+0.5*d;
b=L-a;
if(c~=2&c~=3) error('Boundary Condition Fail'); end;
if(c==2)
        s1=0;
        s3=-q*b*d/(8*L^2)*(4*a^2+8*a*b-d^2);
        s4=0;
        s6=0;
        s2=-q*d*(L-a)/L + 1/L*(s3+s6);
        s5=-q*d-s2;
        so=[s1;s2;s3;s4;s5;s6];
end;
if(c==3)
        s1=0;
        s3=0;
        s4=0;
        s6=q*b*d/(8*L^2)*(4*a^2+8*a*b-d^2);
        s5=-q*d*(L-b)/L-1/L*(s3+s6);
        s2=-q*d-s5;
        so=[s1;s2;s3;s4;s5;s6];
end;
```

3.4.12. Program fepm.m

Program fepm.m berfungsi untuk menghitung gaya jepit ujung batang termodifikasi oleh beban terpusat. Program fepm.m adalah sebagai berikut ini.

```
function so=fepm(p1,x1,L,c)
% fepm   produces fixed end forces due to  load
%        so = fepm(p1,x1,L,c)
%               so  = fixed end forces
%               p1  = point load
%               x1  = p1 distance from 1st node
%           L    = length of the member
%               c   = 2  |--------o
%                   = 3  o--------|
% see also: FEQM.M, FEQ.M, FEG.M, FET.M, FEP.M, FEQM.M, FETM.M

% developed Yoyong Arfiadi
% 9611-- 00
% 130720
% Atma Jaya Yogyakarta University

if(c~=2&c~=3); error('Boundary Condition Fail'); end;
if(x1>L | x1 <0); error('Load Not on the Span');end;
if (c==2)
        s1=0;
        s3=-p1*x1*(L-x1)/(2*L^2)*(2*L-x1);
        s4=0;
        s6=0;
        s2=-(1/L*(p1*(L-x1))) + (s3+s6)/L;
        s5=-(1/L*(p1*x1)) - (s3+s6)/L;
        so=[s1;s2;s3;s4;s5;s6];
end;
if(c==3)
      x2=L-x1;
        s1=0;
        s3=0;
        s4=0;
        s6=p1*x2*(L-x2)/(2*L^2)*(2*L-x2);
        s2=-(1/L*(p1*(L-x1))) + (s3+s6)/L;
        s5=-(1/L*(p1*x1)) - (s3+s6)/L;
        so=[s1;s2;s3;s4;s5;s6];
end;
```

3.4.13. Program fegm.m

Program fegm.m berfungsi untuk menghitung gaya jepit ujung batang termodifikasi oleh beban segitiga. Program fegm.m adalah sebagai berikut ini.

```
function so=fegm(q,L,c)
% fegm   produces fixed end forces due to triangular load
%        for modified member
%        so  = fegm(q,L,c)
%            so  = fixed end forces
%            q   = triangular load intensity
%            L   = length of the member
%            c   = 2   |----------o
%                  3   o----------|
% see also: FEP.M, FEQ.M, FET.M,FEPM.M,FEQM.M,FETM.M

% developed Yoyong Arfiadi
% 9611-- 0.0
% 130720 v1.0
% Atma Jaya Yogyakarta University

if(c~=2 & c~=3) error('Boundary Condition Fail');end;
if (c==2)
        s1=0;
        s3=-5/64*q*L^2;
        s4=0;
        s6=0;
        s2=-0.25*q*L+1/L*(s3+s6);
        s5=-0.25*q*L-1/L*(s3+s6);
        so=[s1;s2;s3;s4;s5;s6];
end;
if(c==3)
        s1=0;
        s3=0;
        s4=0;
        s6=5/64*q*L^2;
        s2=-0.25*q*L+1/L*(s3+s6);
        s5=-0.25*q*L-1/L*(s3+s6);
        so=[s1;s2;s3;s4;s5;s6];
end;
```

3.4.14. Program fetm.m

Program fetm.m berfungsi untuk menghitung gaya jepit ujung batang termodifikasi oleh beban trapesium. Program fetm.m adalah sebagai berikut ini.

```
function so=fetm(q,a,L,c)
% fetm.m  produces fixed end forces due to
%       trapezoidal load
%       so = fetm(q,a,L)
%           so = fixed end forces
%           q  = load intensity
%               a  = distance from first node
%                    to the first peak load
%               L  = length of the member
```

```
%                c  = 2 |------0
%                   = 3 0------|
% see also: FEQ.M, FEP.M, FEG.M, FEPM.M, FEGM.M, FEQM.M, FET.M

% developed Yoyong Arfiadi
% 9611-- 0.0
% 130720 v1.0
% Atma Jaya Yogyakarta University

if(2*a>L) error('Not a Trapezoidal Load');end;
if(c==2)
        s1=0;
        s3=-(q/(8*L))*(L^3-a^2*(2*L-a));
        s4=0;
        s6=0;
        s2=-(0.5*q*a + 0.5*q*(L-2*a)) + 1/L*(s3+s6);
        s5=-(0.5*q*a + 0.5*q*(L-2*a)) - 1/L*(s3+s6);
        so=[s1;s2;s3;s4;s5;s6];
end;
if(c==3)
        s1=0;
        s3=0;
        s4=0;
        s6=(q/(8*L))*(L^3-a^2*(2*L-a));
        s2=-(0.5*q*a + 0.5*q*(L-2*a)) + 1/L*(s3+s6);
        s5=-(0.5*q*a + 0.5*q*(L-2*a)) - 1/L*(s3+s6);
        so=[s1;s2;s3;s4;s5;s6];
end;
```

3.4.15. Program feg2m.m

Program feg2m.m berfungsi untuk menghitung gaya jepit ujung batang termodifikasi oleh dua buah beban segitiga. Program feg2m,m adalah sebagai berikut ini.

```
function so=feg2m(q,L,c)
% feg2m  produces fixed end forces due to 2 symmetric
%        triangular load
%        so = feg2m(q,L)
%               so = fixed end forces
%               q   = triangular load intensity
%               L   = length of the member
% see also: FEP.M, FEP.M, FEQ.M, FEPM.M, FEQM.M, FEG2.M

% developed Yoyong Arfiadi
% 9611-- 0.0
% 130720 v1.0
% Atma Jaya Yogyakarta University
```

```
if(c~=2&c~=3);error('Boundary Condition Fail');end;
if(c==2)
        s1=0;
        s3=-17/256*q*L^2;
        s4=0;
        s6=0;
        s2=-0.25*q*L + (s3+s6)/L;
        s5=-0.25*q*L - (s3+s6)/L;
        so=[s1;s2;s3;s4;s5;s6];
end;
if(c==3)
        s1=0;
        s3=0;
        s4=0;
        s6=17/256*q*L^2;
        s2=-0.25*q*L + (s3+s6)/L;
        s5=-0.25*q*L - (s3+s6)/L;
        so=[s1;s2;s3;s4;s5;s6];
end;
```

3.4.16. Program peqj.m

Program peqj.m berfungsi untuk menghitung beban titik ekivalen berdasarkan gaya jepit ujung dan vektor tujuan batang yang ditinjau. Program peqj.m adalah sebagai berikut ini.

```
function P = peqj(so,T,id,tdf)
% peqj.m produces equivalent joint load for a specified member
%        in terms of total degree of freedom of the structure
%        P = peqj(so,T,id,tdf)
%               P  = equivalent joint load vector (tdf X 1)
%               so = member fixed end forces
%               T  = transformation matrix
%               id = destination vector
%               tdf= total degree of freedom of the structure

% developed Yoyong Arfiadi
% 961126 0.0
% 130720
% Atma Jaya Yogyakarta University

P=zeros(tdf,1);
Po=T'*so;
for i=1:6
      ii=id(1,i);
        if(ii~=0)
        P(ii,1)=-Po(i,1);
      end
end
```

3.4.17. Program dissf.m

Program dissf.m berfungsi untuk menghitung deformasi batang, jika prpindahan global struktur sudah dihitung. Program dissf.m adalah sebagai berikut ini.

```
function v=dissf(r,id,T)
% dissf.m   calculates displacements of the member
%           in local coordinate  for plane frame
%           v = dissf(r,id,T)
%                 v  = member deformations
%                 r  = degree of freedom
%                 id = destination vector of the member
%                 T  = member transformation matrix
% see also DISST.M

% developed Yoyong Arfiadi
% 9611-- 0.0
% 0512-- change of 'dist'
% 130720 v1.0
% Atma Jaya Yogyakarta University

for i=1:6
   ii =id(1,i);
   if (ii == 0);
      rl(i,1) = 0;
   else
      rl(i,1)=r(ii,1);
  end;
end;
v=T*rl;
```

3.4.18. Program stref.m

Program stref.m berfungsi untuk menghitung gaya batang. Program stref.m adalah sebagai berikut ini.

```
function s=stref(k,v,so)
% stref.m   produces internal forces for plane frame structures
%           s=stref(k,v,so)
%                 s  = internal forces
%                 k  = member stiffness matrix in local coordinate
%                 v  = member deformation vector
%                 so = fixed end forces

%           developed Yoyong Arfiadi
%           961126 0.0
%       130720
```

```
% Atma Jaya Yogyakarta University

s=k*v + so;
```

3.4.19. Program klfg.m

Program klfg.m berfungsi untuk membentuk matriks kekakuan batang dengan daerah kaku pada ujung batang, baik matriks kekakuan untuk bagian dalam daerah kaku dan untuk jarak pusat ke pusat batang yang membatasi. Program ini juga membentuk matriks transformasi untuk mentransformasikan matriks kekakuan bagian dalam (tanpa ujung kaku) menjadi matriks kekakuan pusat ke pusat. Program klfg.m adalah sebagai berikut ini.

```
function [k,kpr,Tpr]=klfg(E,A,I,Lo,f,v,ri,rj)
% KLFG generates member stiffness matrix with shear deformation and rigid zone
%      for plane frame in local coordinate
%      [k, kpr,Tpr] = KLFG(E,A,I,Lo,f,v,ri,rj)
%             k = member stiffness matrix for inner part
%            kpr = member stiffness matrix for center line
%            Tpr = transformation matrix to transform k to kpr
%             E = modulus of elasticity
%             A = area
%             I = second moment area
%             Lo = length of the member center to center
%            f = shear area reduction
%            f = A/Abar
%            Abar = effective shear area
%            v = Poisson's ratio
%            ri = length of rigid i-end
%            rj = length of rigid j-end
% see also: KLFM.M, KLT.M, KLF.M, KLFS.M

% developed Yoyong Arfiadi
% Atma Jaya Yogyakarta University
%      960925(klf)
%   050902  klfs
%   0807-- 0 klfg
%   130805 ---

G=E/(2*(1+v));

L=Lo-(ri+rj);  %%

a0=E*A/L;

a=12*f*E*I/(G*A*L^2);

k=[a0      0        0      -a0        0        0;
   0     12*E*I/((1+a)*L^3)   6*E*I/((1+a)*L^2) 0 -12*E*I/((1+a)*L^3) 6*E*I/((1+a)*L^2);
   0    6*E*I/((1+a)*L^2) (4+a)*E*I/((1+a)*L) 0 -6*E*I/((1+a)*L^2) (2-a)*E*I/((1+a)*L)
  -a0      0        0      a0         0        0;
   0    -12*E*I/((1+a)*L^3)   -6*E*I/((1+a)*L^2) 0 12*E*I/((1+a)*L^3) -
6*E*I/((1+a)*L^2);
```

```
  0     6*E*I/((1+a)*L^2) (2-a)*E*I/((1+a)*L) 0 -6*E*I/((1+a)*L^2)
(4+a)*E*I/((1+a)*L)];

Tpr=[1  0  0  0  0  0
     0  1  ri 0  0  0
     0  0  1  0  0  0
     0  0  0  1  0  0
     0  0  0  0  1  -rj
     0  0  0  0  0  1];

 kpr=Tpr'*k*Tpr;
```

3.4.19. Rekapitulasi program untuk portal bidang

Seluruh program yang digunakan dalam analisis portal bidang dapat dilihat pada Tabel 3.2.

Tabel 3.2. Subprogram untuk portal bidang

NO	PROGRAM	FUNGSI	*OUTPUT*	*INPUT*	*SYNTAX*
1	coor.m	Menghasilkan koordinat titik nodal	Ni(xi,yi)	xi, yi	Ni=coor(xi,yi)
2	memf.m	Menghitung panjang batang dan matriks transformasi portal bidang	L, T	ci, cj	[L,T]=memf(ci,cj)
3	klf.m	Menghasilkan matriks kekakuan batang portal bidang dalam koordinat lokal	km	Em, Am, Im, Lm	km=klf(Em,Am,Im,Lm)
4	klfm.m	Menghasilkan matriks kekakuan batang termodifikasi suatu portal bidang	km	Em, Am, Im, Lm, C	km=klfm(Em,Am,Im,Lm,C)
5	assf.m	Menggabungkan matriks kekakuan batang dalam koordinat lokal	Kam	Km, Idm, TDF	Kam=assf(Km,Idm,TDF)
6	feq.m	Menghasilkan gaya ujung jepit akibat beban terbagi rata	So	q, a1, a2, L	So=feq(q,a1,a2,L)
7	fep.m	Menghasilkan gaya ujung jepit akibat beban terpusat	So	P1, x1, L	So=fep(P1,x1,L)

NO	PROGRAM	FUNGSI	*OUTPUT*	*INPUT*	*SYNTAX*
8	fep3.m	Menghasilkan gaya ujung jepit akibat 3 beban terpusat	So	P1, x1, P2, x2, P3, x3, L	So=fep3(P1,x1,P2,x2,P3,x3,L)
9	feg.m	Menghitung gaya ujung jepit akibat beban segitiga	So	q, L	So=feg(q,L)
10	fet.m	Meghitung gaya ujung jepit akibat beban trapesium	So	q, a, L	So=fet(q,a,L)
11	feg2.m	Menghitung gaya ujung jepit akibat 2 beban segitiga simetrik	So	q, L	So=feg2(q,L)
12	feqm.m	Menghitung gaya ujung jepit akibat beban terbagi rata pada batang termodifikasi	So	q, a1, a2, L, c	So=feqm(q,a1,a2,L,c)
13	fepm.m	Menghitung gaya ujung jepit akibat beban terpusat pada batang termodifikasi	So	P1, x1, L, c	So=fepm(P1,x1,L,c)
14	fegm.m	Menghitung gaya ujung jepit akibat beban segitiga pada batang termodifikasi	So	q, L, c	So=fegm(q,L,c)
15	fetm.m	Meghitung gaya ujung jepit akibat beban trapesium pada batang termodifikasi	So	q, a, L, c	So=fetm(q,a,L,c)
16	feg2m.m	Menghitung gaya ujung jepit akibat 2 beban segitiga simetrik pada batang termodifikasi	So	q, L, c	So=feg2m(q,L,c)
17	peqj.m	Menghitung beban titik ekivalen suatu batang dan menempatkannya pada vektor beban luar	Rm	So, T, ID, TDF	Rm=peqj(So,T,ID,TDF)

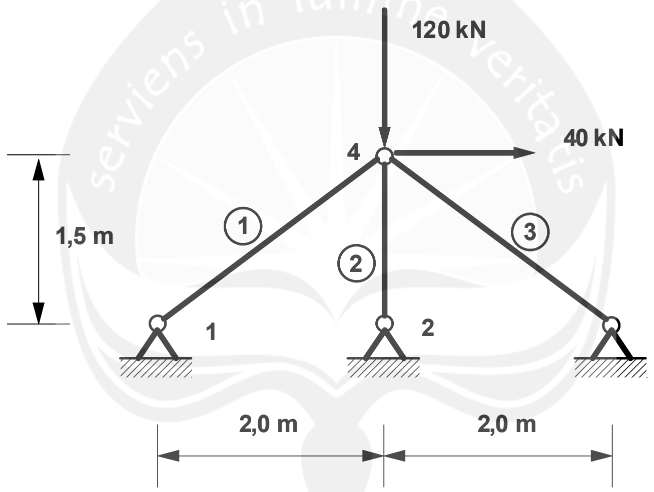
NO	PROGRAM	FUNGSI	*OUTPUT*	*INPUT*	*SYNTAX*
18	solv.m	Menyelesaikan persamaan linier simultan	U	K, R	U=solv(K,R)
19	dissf.m	Menghitung deformasi batang suatu portal bidang	u	U, ID, T	u=disf(U,ID,T)
20	stref.m	Menghitung gaya-gaya ujung batang pada portal bidang	S	k, u, So	S=stref(k,u,So)
21	plot2d.m	Menggambar struktur 2D	gambar	N, linestyle, linewidth	figure = plot2d(N,linestyle,linewidth,t)
22	Klfg.m	Membentuk matriks kekakuan bagian dalam (k), matriks kekakuan pusat ke pusat (kpr), dan matriks transformasi untuk mentransformasikan k ke kpr			[k,kpr,Tpr]=klfg(E,A,I,Lo,f,v,ri,rj)

BAB IV
APLIKASI PROGRAM

4.1. APLIKASI PADA STRUKTUR RANGKA BIDANG

4.1.1. Contoh 1 struktur rangka bidang

Sebagai contoh pertama ditinjau struktur rangka bidang seperti ditunjukkan pada Gambar 4.1. Struktur diambil dari Arfiadi (2011). E = 200 kN/mm^2, A_1 = 2500 mm^2, A_2 = 1800 mm^2, A_3 = A_1. Untuk contoh pertama ini, langkah hitungan manual akan disajikan dulu agar lebih jelas. Langkah hitungan diambil dari Arfiadi (2011). Setelah hitungan manual, disajikan program FreeMat untuk penyelesaian kasus yang sama.

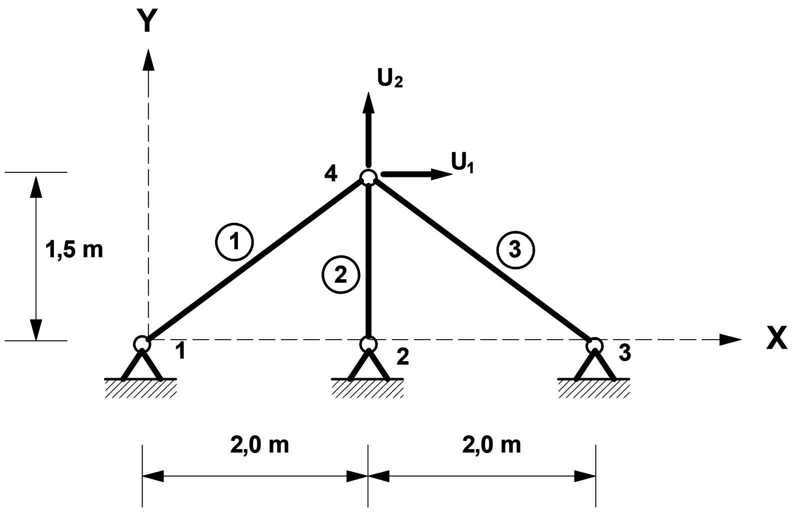


serviens in lumine veritatis

Gambar 4.1. Contoh 1 rangka bidang

Penyelesaian secara manual:

Derajat kebebasan struktur = 2 seperti ditunjukkan pada Gambar 3.21. Hubungan antara batang ditetapkan seperti pada Tabel 4.1.



Gambar 4.2. Derajat kebebasan contoh 1 rangka bidang

Tabel 4.1. Hubungan antar batang contoh 1 rangka bidang

Batang	Titik-i	Titik-j
1	1	4
2	2	4
3	4	3

Nilai-nilai $\frac{EA}{L}$ untuk masing-masing batang selanjutnya dihitung dan disajikan dalam Tabel 4.2.

Tabel 4.2. Sifat-sifat batang contoh 1 rangka bidang

Batang	E (kN/mm^2)	A (mm^2)	L (mm)	$\frac{EA}{L}$ (kN/mm)
1	200	2500	2500	200
2	200	1800	1500	240
3	200	2500	2500	200

Matriks kekakuan, matriks transformasi dan vektor tujuan masing-masing batang:

- **Batang 1:**

Orientasi batang:

$$c = \cos\alpha = \frac{X_j - X_i}{L} = \frac{2000-0}{2500} = 0{,}8,$$

$$s = \sin\alpha = \frac{Y_j - Y_i}{L} = \frac{1500-0}{2500} = 0{,}6.$$

Matriks transformasi:

$$[T]_1 = \begin{bmatrix} c & s & 0 & 0 \\ -s & c & 0 & 0 \\ 0 & 0 & c & s \\ 0 & 0 & -s & c \end{bmatrix} = \begin{bmatrix} 0{,}8 & 0{,}6 & 0 & 0 \\ -0{,}6 & 0{,}8 & 0 & 0 \\ 0 & 0 & 0{,}8 & 0{,}6 \\ 0 & 0 & -0{,}6 & 0{,}8 \end{bmatrix}.$$

Matriks kekakuan batang dalam koordinat lokal:

$$[k]_1 = \begin{bmatrix} \frac{EA}{L_1} & 0 & -\frac{EA}{L_1} & 0 \\ 0 & 0 & 0 & 0 \\ -\frac{EA}{L_1} & 0 & \frac{EA}{L_1} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 200 & 0 & -200 & 0 \\ 0 & 0 & 0 & 0 \\ -200 & 0 & 200 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \text{kN/mm.}$$

Matriks kekakuan batang dalam koordinat global:

$$[K]_1 = \frac{EA}{L_m} \begin{bmatrix} c^2 & sc & -c^2 & -sc \\ sc & s^2 & -sc & -s^2 \\ -c^2 & -sc & c^2 & sc \\ -sc & -s^2 & sc & s^2 \end{bmatrix}$$

$$[K]_1 = \begin{bmatrix} 128 & 96 & -128 & -96 \\ 96 & 72 & -96 & -72 \\ -128 & -96 & 128 & 96 \\ -96 & -72 & 96 & 72 \end{bmatrix}$$

Vektor tujuan:

$$[ID]_1 = [0 \;\; 0 \;\; 1 \;\; 2]$$

- **Batang 2:**

Orientasi batang:

$$c = \cos\alpha = \frac{X_j - X_i}{L} = \frac{2000 - 2000}{1500} = 0,$$

$$s = \sin\alpha = \frac{Y_j - Y_i}{L} = \frac{1500 - 0}{1500} = 1.$$

Matriks transformasi:

$$[T]_2 = \begin{bmatrix} c & s & 0 & 0 \\ -s & c & 0 & 0 \\ 0 & 0 & c & s \\ 0 & 0 & -s & c \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \end{bmatrix}.$$

Matriks kekakuan batang dalam koordinat lokal:

$$[k]_2 = \begin{bmatrix} 240 & 0 & -240 & 0 \\ 0 & 0 & 0 & 0 \\ -240 & 0 & 240 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \text{kN/mm.}$$

Matriks kekakuan batang dalam koordinat global:

$$[K]_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 240 & 0 & -240 \\ 0 & 0 & 0 & 0 \\ 0 & -240 & 0 & 240 \end{bmatrix}$$

Vektor tujuan:

$$[ID]_2 = [0 \;\; 0 \;\; 1 \;\; 2]$$

- **Batang 3:**

Orientasi batang:

$$c = \cos\alpha = \frac{X_j - X_i}{L} = \frac{4000-2000}{2500} = 0{,}8,$$

$$s = \sin\alpha = \frac{Y_j - Y_i}{L} = \frac{0-1500}{2500} = -0{,}6.$$

Matriks transformasi:

$$[T]_3 = \begin{bmatrix} c & s & 0 & 0 \\ -s & c & 0 & 0 \\ 0 & 0 & c & s \\ 0 & 0 & -s & c \end{bmatrix} = \begin{bmatrix} 0{,}8 & -0{,}6 & 0 & 0 \\ 0{,}6 & 0{,}8 & 0 & 0 \\ 0 & 0 & 0{,}8 & -0{,}6 \\ 0 & 0 & 0{,}6 & 0{,}8 \end{bmatrix}.$$

Matriks kekakuan batang dalam koordinat lokal:

$$[k]_3 = \begin{bmatrix} 200 & 0 & -200 & 0 \\ 0 & 0 & 0 & 0 \\ -200 & 0 & 200 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \text{kN/mm.}$$

Matriks kekakuan batang dalam koordinat global:

$$[K]_3 = \begin{bmatrix} 128 & -96 & -128 & 96 \\ -96 & 72 & 96 & -72 \\ -128 & 96 & 128 & -96 \\ 96 & -72 & -96 & 72 \end{bmatrix}$$

Vektor tujuan:

$$[ID]_3 = [1 \;\; 2 \;\; 0 \;\; 0]$$

<u>Pembentukan matriks kekakuan struktur:</u>

Matriks kekakuan dirakit dengan memperhatikan masing-masing vektor tujuannya.

Batang 1:

$$\begin{array}{cc} [ID]_1 \rightarrow & \begin{array}{cccc} 0 & 0 & 1 & 2 \end{array} \\ [K]_1 = & \begin{bmatrix} 128 & 96 & -128 & -96 \\ 96 & 72 & -96 & -72 \\ -128 & -96 & 128 & 96 \\ -96 & -72 & 96 & 72 \end{bmatrix} \begin{array}{c} 0 \\ 0 \\ 1 \\ 2 \end{array} \\ & \uparrow \\ & [ID]_1 \end{array}$$

Batang 2:

$$\begin{array}{cc} [ID]_2 \rightarrow & \begin{array}{cccc} 0 & 0 & 1 & 2 \end{array} \\ [K]_2 = & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 240 & 0 & -240 \\ 0 & 0 & 0 & 0 \\ 0 & -240 & 0 & 240 \end{bmatrix} \begin{array}{c} 0 \\ 0 \\ 1 \\ 2 \end{array} \\ & \uparrow \\ & [ID]_2 \end{array}$$

Batang 3:

$$
\begin{array}{l}
[\mathrm{ID}]_3 \rightarrow \quad 1 \quad\quad 2 \quad\quad 0 \quad\quad 0 \\
[\mathrm{K}]_3 = \begin{array}{c} \begin{bmatrix} 128 & -96 & -128 & 96 \\ -96 & 72 & 96 & -72 \\ -128 & 96 & 128 & -96 \\ 96 & -72 & -96 & 72 \end{bmatrix} \end{array} \begin{array}{c} 1 \\ 2 \\ 0 \\ 0 \end{array}
\end{array}
$$

$$\uparrow$$

$$[\mathrm{ID}]_3$$

Dengan merakit matriks kekakuan masing-masing batang dalam koordinat global sesuai dengan vektor tujuannya diperoleh matriks kekakuan struktur:

$$[\mathrm{K}] = \begin{bmatrix} 256 & 0 \\ 0 & 384 \end{bmatrix} \text{ kN/mm}$$

Vektor beban luar:

Vektor beban luar yang sesuai dengan derajat kebebasan struktur adalah

$$\{\mathrm{P}\} = \begin{Bmatrix} \mathrm{P}_1 \\ \mathrm{P}_2 \end{Bmatrix} = \begin{Bmatrix} 40 \\ -120 \end{Bmatrix} \text{ kN.}$$

Penyelesaian persamaan linier:

Persamaan linier

$$[\mathrm{K}]\{\mathrm{U}\} = \{\mathrm{P}\}$$

$$\begin{bmatrix} 256 & 0 \\ 0 & 384 \end{bmatrix} \begin{Bmatrix} \mathrm{U}_1 \\ \mathrm{U}_2 \end{Bmatrix} = \begin{Bmatrix} 40 \\ -120 \end{Bmatrix}$$

Karena bentuk akhir matriks kekakuan struktur berupa matriks diagonal, perpindahan U_1, dan U_2 dapat diperoleh dengan mudah yaitu:

U_1 =40/256 = 0,15625 mm,

U_2 = -120/384 = − 0,3125 mm.

Vektor perpindahan struktur menjadi

$$\begin{Bmatrix} U_1 \\ U_2 \end{Bmatrix} = \begin{Bmatrix} 0{,}15625 \\ -0{,}3125 \end{Bmatrix} \text{mm.}$$

<u>Gaya batang dan deformasi batang:</u>

- **Batang 1:**

Perpindahan dalam koordinat lokal (deformasi):

$$\{u\}_1 = [T]_1 \{U\}_1$$

$$\begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{Bmatrix}_1 = \begin{Bmatrix} u_{xi} \\ u_{yi} \\ u_{xj} \\ u_{yj} \end{Bmatrix}_1 = \begin{bmatrix} 0{,}8 & 0{,}6 & 0 & 0 \\ -0{,}6 & 0{,}8 & 0 & 0 \\ 0 & 0 & 0{,}8 & 0{,}6 \\ 0 & 0 & -0{,}6 & 0{,}8 \end{bmatrix} \begin{Bmatrix} 0 \\ 0 \\ U_1 \\ U_2 \end{Bmatrix}$$

$$\begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{Bmatrix}_1 = \begin{Bmatrix} u_{xi} \\ u_{yi} \\ u_{xj} \\ u_{yj} \end{Bmatrix}_1 = \begin{bmatrix} 0{,}8 & 0{,}6 & 0 & 0 \\ -0{,}6 & 0{,}8 & 0 & 0 \\ 0 & 0 & 0{,}8 & 0{,}6 \\ 0 & 0 & -0{,}6 & 0{,}8 \end{bmatrix} \begin{Bmatrix} 0 \\ 0 \\ 0{,}15625 \\ -0{,}3125 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ -0{,}0625 \\ -0{,}34375 \end{Bmatrix} \text{mm}$$

Gaya batang:

$$\{S\}_1 = [k]_1 \{u\}_1$$

$$\begin{Bmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \end{Bmatrix}_1 = \begin{Bmatrix} S_{xi} \\ S_{yi} \\ S_{xj} \\ S_{yj} \end{Bmatrix}_1 = \begin{bmatrix} 200 & 0 & -200 & 0 \\ 0 & 0 & 0 & 0 \\ -200 & 0 & 200 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} 0 \\ 0 \\ -0{,}0625 \\ -0{,}34375 \end{Bmatrix} = \begin{Bmatrix} 12{,}5 \\ 0 \\ -12{,}5 \\ 0 \end{Bmatrix} \text{kN}$$

- **Batang 2:**

Perpindahan dalam koordinat lokal (deformasi):

$$\{u\}_2 = [T]_2 \{U\}_2$$

$$\begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{Bmatrix}_2 = \begin{Bmatrix} u_{xi} \\ u_{yi} \\ u_{xj} \\ u_{yj} \end{Bmatrix}_{21} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \end{bmatrix} \begin{Bmatrix} 0 \\ 0 \\ U_1 \\ U_2 \end{Bmatrix}$$

$$\begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{Bmatrix}_2 = \begin{Bmatrix} u_{xi} \\ u_{yi} \\ u_{xj} \\ u_{yj} \end{Bmatrix}_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \end{bmatrix} \begin{Bmatrix} 0 \\ 0 \\ 0{,}15625 \\ -0{,}3125 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ -0{,}3125 \\ -0{,}15625 \end{Bmatrix} \text{mm}$$

Gaya batang:

$$\{S\}_2 = [k]_2 \{u\}_2$$

$$\begin{Bmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \end{Bmatrix}_2 = \begin{Bmatrix} S_{xi} \\ S_{yi} \\ S_{xj} \\ S_{yj} \end{Bmatrix}_2 = \begin{bmatrix} 240 & 0 & -240 & 0 \\ 0 & 0 & 0 & 0 \\ -240 & 0 & 240 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} 0 \\ 0 \\ -0{,}3125 \\ -0{,}15625 \end{Bmatrix} = \begin{Bmatrix} 75 \\ 0 \\ -75 \\ 0 \end{Bmatrix} \text{kN}$$

- **Batang 3:**

Perpindahan dalam koordinat lokal (deformasi):

$$\{u\}_3 = [T]_3 \{U\}_3$$

$$\begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{Bmatrix}_3 = \begin{Bmatrix} u_{xi} \\ u_{yi} \\ u_{xj} \\ u_{yj} \end{Bmatrix}_3 = \begin{bmatrix} 0{,}8 & -0{,}6 & 0 & 0 \\ 0{,}6 & 0{,}8 & 0 & 0 \\ 0 & 0 & 0{,}8 & -0{,}6 \\ 0 & 0 & 0{,}6 & 0{,}8 \end{bmatrix} \begin{Bmatrix} U_1 \\ U_2 \\ 0 \\ 0 \end{Bmatrix}$$

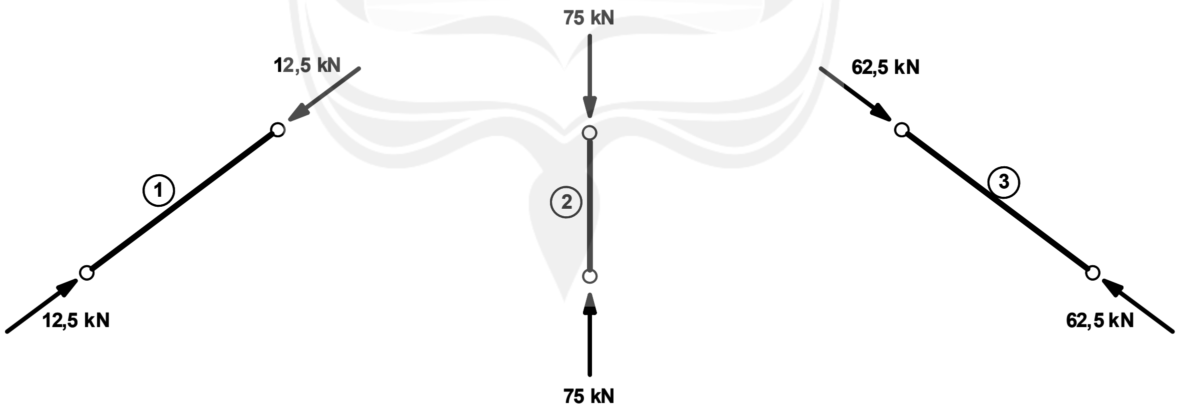
$$\begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{Bmatrix}_3 = \begin{Bmatrix} u_{xi} \\ u_{yi} \\ u_{xj} \\ u_{yj} \end{Bmatrix}_3 = \begin{bmatrix} 0,8 & -0,6 & 0 & 0 \\ 0,6 & 0,8 & 0 & 0 \\ 0 & 0 & 0,8 & -0,6 \\ 0 & 0 & 0,6 & 0,8 \end{bmatrix} \begin{Bmatrix} 0,15625 \\ -0,3125 \\ 0 \\ 0 \end{Bmatrix} = \begin{Bmatrix} 0,3125 \\ -0,15625 \\ 0 \\ 0 \end{Bmatrix} \text{mm}$$

Gaya batang:

$$\{S\}_3 = [k]_3 \{u\}_3$$

$$\begin{Bmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \end{Bmatrix}_3 = \begin{Bmatrix} S_{xi} \\ S_{yi} \\ S_{xj} \\ S_{yj} \end{Bmatrix}_3 = \begin{bmatrix} 200 & 0 & -200 & 0 \\ 0 & 0 & 0 & 0 \\ -200 & 0 & 200 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} 0,3125 \\ -0,15625 \\ 0 \\ 0 \end{Bmatrix} = \begin{Bmatrix} 62,5 \\ 0 \\ -62,5 \\ 0 \end{Bmatrix} \text{kN}$$

Diagram benda bebas ditunjukkan pada Gambar 4.3.



Gambar 4.3. Diagram benda bebas contoh 1 rangka bidang

Penyelesaian dengan FreeMat:

Untuk menyelesaikan contoh ini, dibuat berkas m (ekstensi m pada berkas adalah *default* dari FreeMat). Pada *command window*, pilih menu File →New File sehingga tampil jendela baru yaitu jendela Editor tempat kita menulis berkas. Lalu ketik program berikut ini dan simpan dengan nama c1_tr.m. Dalam hal ini satuan panjang adalah mm dan satuan gaya adalah kN.

```
%contoh 1 struktur rangka bidang

n1=coor(0,0);       %--- data koordinat
n2=coor(2000,0);
n3=coor(4000,0);
n4=coor(2000,1500);

E=200;              %--- data bahan dan tampang
A1=2500;
A2=1800;
A3=A1;

[L1,T1]=memt(n1,n4) %--- menghitung L dan T
k1=klt(E,A1,L1)     %--- membentuk k lokal
K1=kg(k1,T1)        %--- membentuk K dlm koordinat global = T' k T
ID1=[0 0 1 2]       %--- vektor tujuan

[L2,T2]=memt(n2,n4)
k2=klt(E,A2,L2)
K2=kg(k2,T2)
ID2=[0 0 1 2]

[L3,T3]=memt(n4,n3)
k3=klt(E,A3,L3)
K3=kg(k3,T3)
ID3=[1 2 0 0]

dof=2;               %--- derajat kebebasan struktur

K=asst(K1,ID1,dof);  %--- kontribusi batang 1 terhadap K struktur
K=K+asst(K2,ID2,dof);
K=K+asst(K3,ID3,dof) %--- K struktur

P=[40;-120]          %--- vektor beban

U=solv(K,P)          %--- perpindahan struktur

u1=disst(U,ID1,T1)   %--- deformasi batang: u1 = T1 U1
S1=stret(k1,u1)      %--- gaya batang S = k u

u2=disst(U,ID2,T2)
S2=stret(k2,u2)

u3=disst(U,ID3,T3)
S3=stret(k3,u3)
```

Tanda persen (%) pada berkas menunjukkan komentar, yaitu karakter atau tulisan setelah tanda % tidak akan diproses. Selain itu, pada setiap baris program yang tidak diakhiri dengan tanda titik koma (;) berarti hasil analisis akan ditampilkan pada layar. Sedangkan baris program yang diakhiri dengan tanda titik koma berarti hasil analisis tidak ditampilkan pada layar.

Setelah berkas disimpan, kita kembali ke *command window* (jendela perintah), lalu ketik nama berkas tanpa ekstensi m, sehingga dihasilkan sebagai berikut ini.

```
--> c1_tr
L1 =
 2500
T1 =
    0.8000    0.6000         0         0
   -0.6000    0.8000         0         0
         0         0    0.8000    0.6000
         0         0   -0.6000    0.8000
k1 =
  200    0 -200    0
    0    0    0    0
 -200    0  200    0
    0    0    0    0
K1 =
  128   96 -128  -96
   96   72  -96  -72
 -128  -96  128   96
  -96  -72   96   72
ID1 =
 0 0 1 2
L2 =
 1500
T2 =
  0  1  0  0
 -1  0  0  0
  0  0  0  1
  0  0 -1  0
k2 =
  240    0 -240    0
    0    0    0    0
 -240    0  240    0
    0    0    0    0
K2 =
    0    0    0    0
    0  240    0 -240
    0    0    0    0
    0 -240    0  240
```

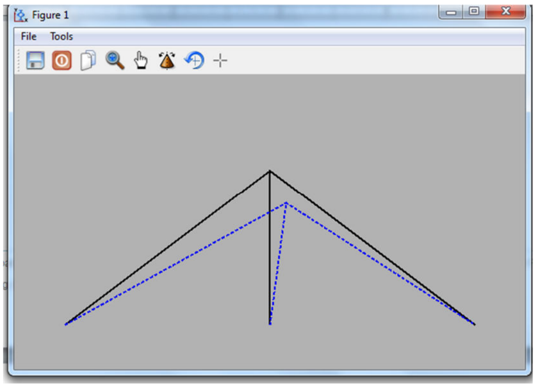
```
ID2 =
 0 0 1 2
L3 =
 2500
T3 =
    0.8000   -0.6000         0         0
    0.6000    0.8000         0         0
         0         0    0.8000   -0.6000
         0         0    0.6000    0.8000
k3 =
  200    0 -200    0
    0    0    0    0
 -200    0  200    0
    0    0    0    0
K3 =
  128  -96 -128   96
  -96   72   96  -72
 -128   96  128  -96
   96  -72  -96   72
ID3 =
 1 2 0 0
K =
 256   0
   0 384
P =
   40
 -120
U =
    0.1562
   -0.3125
u1 =
         0
         0
   -0.0625
   -0.3438
S1 =
   12.5000
         0
  -12.5000
         0
u2 =
         0
         0
   -0.3125
   -0.1562
S2 =
  75
   0
 -75
   0
```

```
u3 =
    0.3125
   -0.1562
         0
         0
S3 =
   62.5000
         0
  -62.5000
         0
-->
```

Jika ingin menggambar geometri struktur dapat digunakan program plot2d, baik dalam keadaan sebelum dibebani mau pun setelah berdeformasi. Dalam hal ini dapat dilakukan langkah berikut ini.

```
M=[n1;n4;n2;n4;n3;n4]          % mesh struktur
plot2d(M,'-k','LineWidth',2)   % menggambar undeformed

sc=1000;                       % fakor skala
Un=[0;0;0;0;0;0;U(1);U(2)]     % perpindahan titik simpul

n1n=n1+sc*Un(1:2)'              % lokasi titik simpul yang baru
n2n=n2+sc*Un(3:4)'
n3n=n3+sc*Un(5:6)'
n4n=n4+sc*Un(7:8)'

Mn=[n1n;n4n;n2n;n4n;n4n;n3n]    % mesh struktur 'deformed'
plot2d(Mn,':b','LineWidth',2)   % menggambar struktur 'deformed'
```

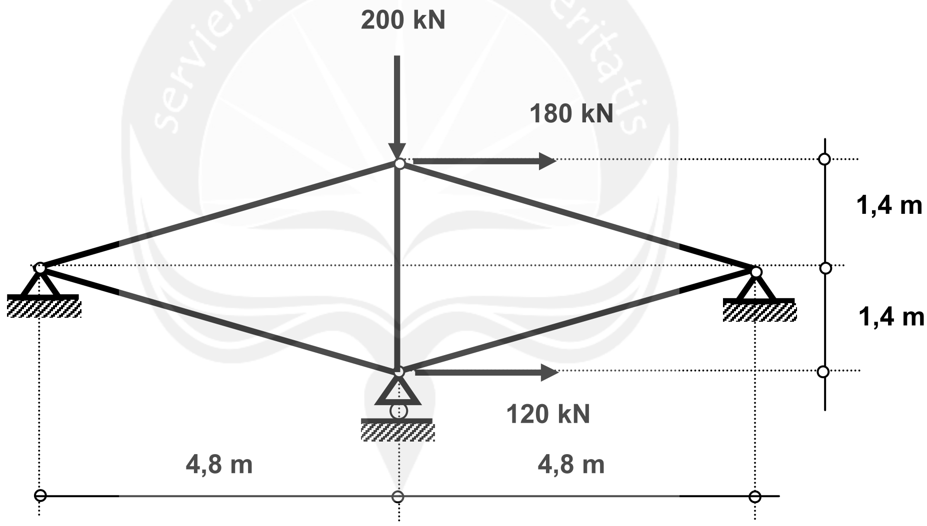
Langkah-langkah tersebut dapat ditulis langsung pada *command window*, atau dibuat suatu berkas dengan nama tertentu, misal plot_c1_tr.m, lalu panggil berkas tersebut dari *command window*. Jika dilakukan akan dihasilkan jendela baru seperti terlihat pada Gambar 4.4.



Gambar 4.4. Tampilan *deformed* dan *undeformed*

4.1.2. Contoh 2 struktur rangka bidang

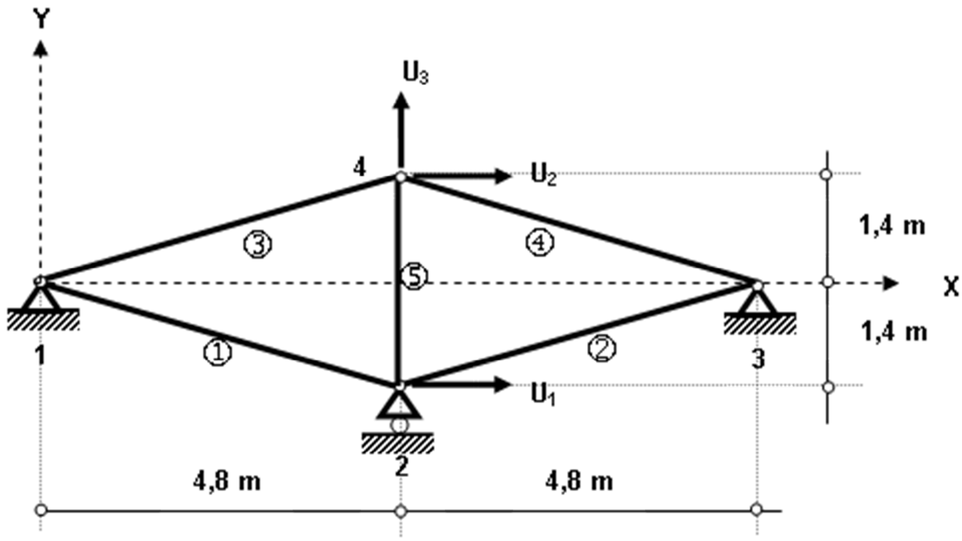
Suatu struktur seperti terlihat pada Gambar 4.5, semua batang mempunyai E = 200 kN/mm^2, A = 2500 mm^2. Hitung gaya-gaya batang dengan metode matriks kekakuan.



Gambar 4.5. Contoh 2 struktur rangka bidang

Penyelesaian degan FreeMat:

Penomoran titik kumpul dan batang dapat dilihat pada Gambar 4.6. Derajat kebebasan diurutkan mulai dari nomor titik kumpul terkecil. Struktur mempunyai 3 derajat kebebasan sehingga ukuran matriks kekakuan struktur adalah 3×3. Hubungan antar batang selanjutnya ditetapkan seperti terlihat pada Tabel 4.3.



Gambar 4.6. Penomoran titik kumpul dan batang, serta derajat kebebasan

Tabel 4.3. Hubungan antar batang contoh 2 rangka bidang.

Batang	Ujung-i	Ujung-j
1	1	2
2	2	3
3	1	4
4	4	3
5	2	4

Selanjutnya dengan Editor FreeMat dibuat berkas dengan nama c2_tr.m sebagai berikut ini. Dalam hal ini diambil satuan panjang dalam mm dan satuan gaya dalam kN.

```
%--- contoh 2 struktur rangka bidang

n1=coor(0,0);
n2=coor(4800,-1400);
n3=coor(9600,0);
n4=coor(4800,1400);

E=200;A=2500;         %--- data bahan

[L1,T1]=memt(n1,n2)  %--- menghitung L dan T
k1=klt(E,A,L1)       %--- k lokal
K1=kg(k1,T1)         %--- K global
id1=[0 0 1 0]        %--- vektor tujuan id
```

```
[L2,T2]=memt(n2,n3)
k2=klt(E,A,L2)
K2=kg(k2,T2)
id2=[1 0 0 0]

[L3,T3]=memt(n1,n4)
k3=klt(E,A,L3)
K3=kg(k3,T3)
id3=[0 0 2 3]

[L4,T4]=memt(n4,n3)
k4=klt(E,A,L4)
K4=kg(k4,T4)
id4=[2 3 0 0]

[L5,T5]=memt(n2,n4)
k5=klt(E,A,L5)
K5=kg(k5,T5)
id5=[1 0 2 3]

tdf=3                   %---total dof

Ks=asst(K1,id1,tdf);    %--- kontribusi batang 1
Ks=Ks+asst(K2,id2,tdf);
Ks=Ks+asst(K3,id3,tdf);
Ks=Ks+asst(K4,id4,tdf);
Ks=Ks+asst(K5,id5,tdf)  %--- Kstruktur

Rs=[120;180;-200];      %--- vektor beban

U=solv(Ks,Rs)           %--- perpindahan

u1=disst(U,id1,T1)      %--- deformasi
S1=stret(k1,u1)         %--- gaya batang

u2=disst(U,id2,T2)
S2=stret(k2,u2)

u3=disst(U,id3,T3)
S3=stret(k3,u3)

u4=disst(U,id4,T4)
S4=stret(k4,u4)

u5=disst(U,id5,T5)
S5=stret(k5,u5)
```

Jika program c2_tr.m dipanggil dari command window dihasilkan sebagai berikut ini.

--> c2_tr lalu tekan *enter*

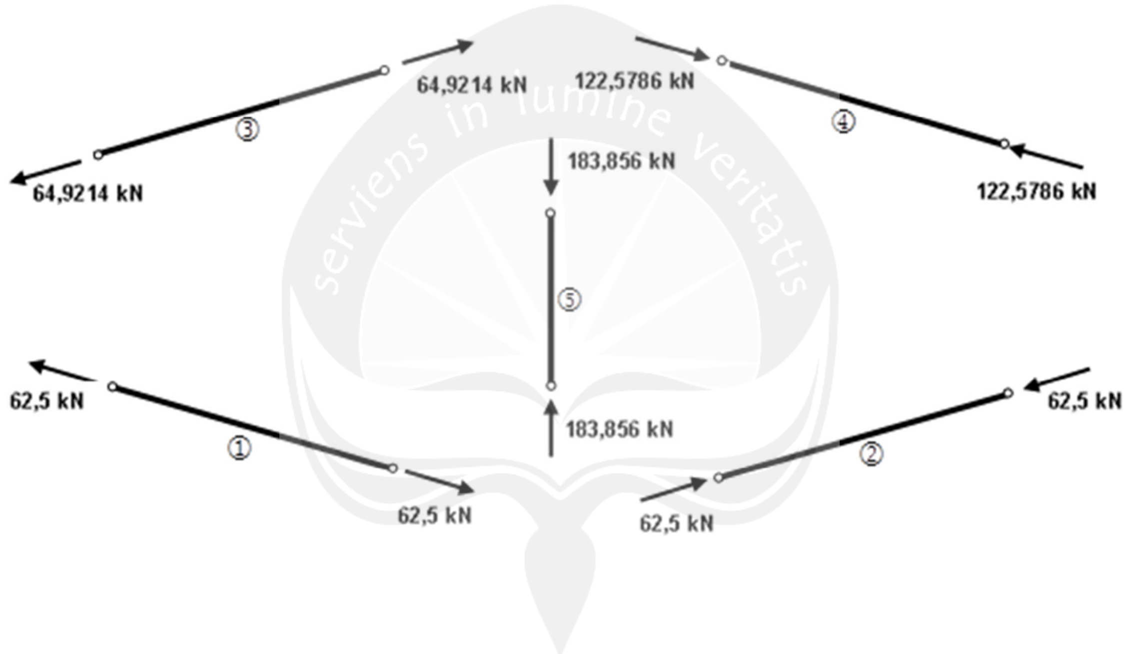
```
L1 =
 5000
T1 =
    0.9600   -0.2800         0         0
    0.2800    0.9600         0         0
         0         0    0.9600   -0.2800
         0         0    0.2800    0.9600
k1 =
  100    0 -100    0
    0    0    0    0
 -100    0  100    0
    0    0    0    0
K1 =
   92.1600  -26.8800  -92.1600   26.8800
  -26.8800    7.8400   26.8800   -7.8400
  -92.1600   26.8800   92.1600  -26.8800
   26.8800   -7.8400  -26.8800    7.8400
id1 =
 0 0 1 0
L2 =
 5000
T2 =
    0.9600    0.2800         0         0
   -0.2800    0.9600         0         0
         0         0    0.9600    0.2800
         0         0   -0.2800    0.9600
k2 =
  100    0 -100    0
    0    0    0    0
 -100    0  100    0
    0    0    0    0
K2 =
   92.1600   26.8800  -92.1600  -26.8800
   26.8800    7.8400  -26.8800   -7.8400
  -92.1600  -26.8800   92.1600   26.8800
  -26.8800   -7.8400   26.8800    7.8400
id2 =
 1 0 0 0
L3 =
 5000
T3 =
    0.9600    0.2800         0         0
   -0.2800    0.9600         0         0
         0         0    0.9600    0.2800
         0         0   -0.2800    0.9600
```

```
k3 =
  100    0 -100    0
    0    0    0    0
 -100    0  100    0
    0    0    0    0
K3 =
   92.1600   26.8800  -92.1600  -26.8800
   26.8800    7.8400  -26.8800   -7.8400
  -92.1600  -26.8800   92.1600   26.8800
  -26.8800   -7.8400   26.8800    7.8400
id3 =
 0 0 2 3
L4 =
 5000
T4 =
    0.9600   -0.2800         0         0
    0.2800    0.9600         0         0
         0         0    0.9600   -0.2800
         0         0    0.2800    0.9600
k4 =
  100    0 -100    0
    0    0    0    0
 -100    0  100    0
    0    0    0    0
K4 =
   92.1600  -26.8800  -92.1600   26.8800
  -26.8800    7.8400   26.8800   -7.8400
  -92.1600   26.8800   92.1600  -26.8800
   26.8800   -7.8400  -26.8800    7.8400
id4 =
 2 3 0 0
L5 =
 2800
T5 =
  0  1  0  0
 -1  0  0  0
  0  0  0  1
  0  0 -1  0
k5 =
  178.5714         0 -178.5714         0
         0         0         0         0
 -178.5714         0  178.5714         0
         0         0         0         0

K5 =
         0         0         0         0
         0  178.5714         0 -178.5714
         0         0         0         0
         0 -178.5714         0  178.5714
id5 =
 1 0 2 3
```

```
tdf =
 3
Ks =
  184.3200         0         0
         0  184.3200         0
         0         0  194.2514
U =
    0.6510
    0.9766
   -1.0296
u1 =
         0
         0
    0.6250
    0.1823
S1 =
  -62.5000
         0
   62.5000
         0
u2 =
    0.6250
   -0.1823
         0
         0
S2 =
   62.5000
         0
  -62.5000
         0
u3 =
         0
         0
    0.6492
   -1.2618
S3 =
  -64.9214
         0
   64.9214
         0

u4 =
    1.2258
   -0.7150
         0
         0
S4 =
  122.5786
         0
 -122.5786
         0
```

```
u5 =
         0
   -0.6510
   -1.0296
   -0.9766
S5 =
  183.8560
         0
 -183.8560
         0
-->
```

Jika gaya batang digambarkan maka diperoleh seperti terlihat pada Gambar 4.7.



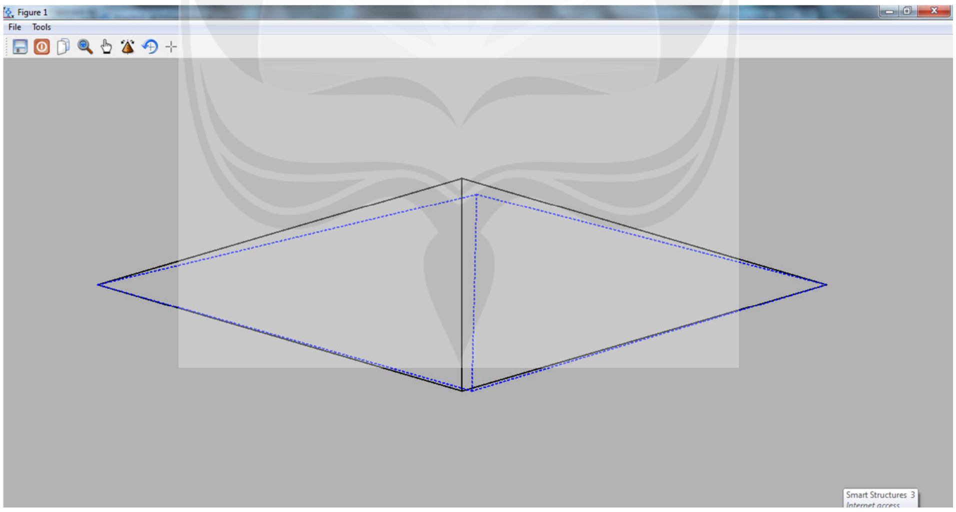
Gambar 4.7. Gaya-gaya batang contoh 2 struktur rangka bidang

Jika geometri struktur ingin digambarkan dapat dilakukan langkah sebagai berikut ini (dapat disimpan dalam suatu berkas tertentu).

```
M=[n1;n2;
n2;n3;
n1;n4;
n4;n3;
n2;n4]                      % mesh ‘undeformed’

plot2d(M,'-k','LineWidth',2) % plot 'undeformed'
```

```
sc=200;                 %faktor skala

Un=[0;0;
U(1);0;
0;0;
U(2);U(3)]              % perpindahan titik simpul

n1n=n1+sc*Un(1:2)'      % letak yang baru
n2n=n2+sc*Un(3:4)'
n3n=n3+sc*Un(5:6)'
n4n=n4+sc*Un(7:8)'

Mn=[n1n;n2n;
n2n;n3n;
n1n;n4n;
n4n;n3n;
n2n;n4n]
plot2d(Mn,':b','LineWidth',2) % plot 'deformed'
```

Jika dijalankan akan diperoleh hasil seperti pada Gambar 4.8.



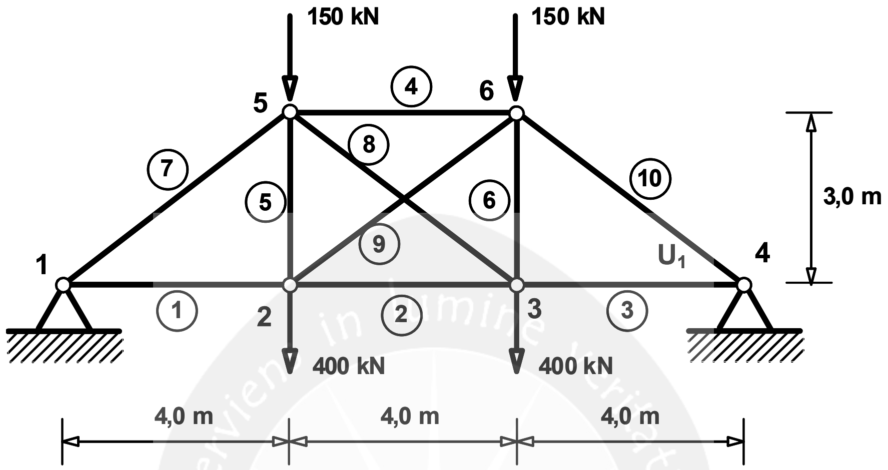
Gambar 4.8. Gambar geometri contorh 2 rangka bidang

4.1.3. Contoh 3 struktur rangka bidang

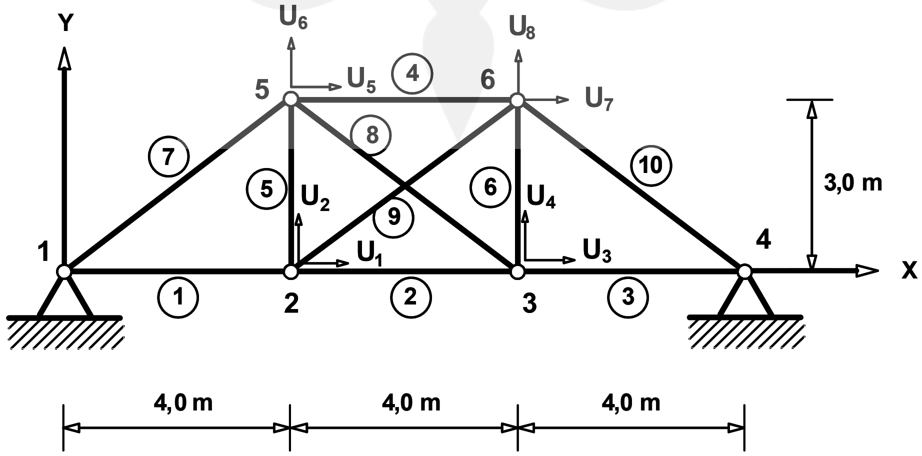
Suatu struktur seperti pada Gambar 4.9 dengan E = 200000 MPa, A = 7392 mm^2 untuk semua batang. Hitung gaya-gaya batang dengan program FreeMat.

Penyelesaian:

Derajat kebebasan struktur = 8 seperti ditunjukkan pada Gambar 4.10. Hubungan antar batang ditetapkan seperti pada Tabel 4.4.



Gambar 4.9. Contoh 3 struktur rangka bidang



Gambar 4.10. Derajat kebebasan stuktur contoh 3 struktur rangka bidang

Tabel 4.4. Hubungan antar batang contoh 3 struktur rangka bidang

Batang	Ujung-i	Ujung-j
1	1	2
2	2	3
3	3	4
4	5	6
5	2	5
6	3	6
7	1	5
8	5	3
9	2	6
10	6	4

Program FreeMat yang disimpan dalam berkas c3_tr.m sebagai berikut ini.

```
%contoh 3 struktur rangka bidang

n1=coor(0,0);                %-- data koordinat
n2=coor(4000,0);
n3=coor(8000,0);
n4=coor(12000,0);
n5=coor(4000,3000);
n6=coor(8000,3000);

E=200;                       %-- kN/mm^2
A=7392;                      %-- m^2

[L1,T1]=memt(n1,n2)          %-- menghitung L dan T
k1=klt(E,A,L1)               %-- k lokal
K1=kg(k1,T1)                 %-- K global
id1=[0 0 1 2];               %-- vektor tujuan

[L2,T2]=memt(n2,n3)
k2=klt(E,A,L2)
K2=kg(k2,T2)
id2=[1 2 3 4];

[L3,T3]=memt(n3,n4)
k3=klt(E,A,L3)
K3=kg(k3,T3)
id3=[3 4 0 0];

[L4,T4]=memt(n5,n6)
k4=klt(E,A,L4)
K4=kg(k4,T4)
id4=[5 6 7 8];
```

```
[L5,T5]=memt(n2,n5)
k5=klt(E,A,L5)
K5=kg(k5,T5)
id5=[1 2 5 6];

[L6,T6]=memt(n3,n6)
k6=klt(E,A,L6)
K6=kg(k6,T6)
id6=[3 4 7 8];

[L7,T7]=memt(n1,n5)
k7=klt(E,A,L7)
K7=kg(k7,T7)
id7=[0 0 5 6];

[L8,T8]=memt(n5,n3)
k8=klt(E,A,L8)
K8=kg(k8,T8)
id8=[5 6 3 4];

[L9,T9]=memt(n2,n6)
k9=klt(E,A,L9)
K9=kg(k9,T9)
id9=[1 2 7 8];

[L10,T10]=memt(n6,n4)
k10=klt(E,A,L10)
K10=kg(k10,T10)
id10=[7 8 0 0];

tdf=8                          %-- jumlah derajat kebebasan

K=asst(K1,id1,tdf);            %-- kontribusi batang 1
K=K+asst(K2,id2,tdf);
K=K+asst(K3,id3,tdf);
K=K+asst(K4,id4,tdf);
K=K+asst(K5,id5,tdf);
K=K+asst(K6,id6,tdf);
K=K+asst(K7,id7,tdf);
K=K+asst(K8,id8,tdf);
K=K+asst(K9,id9,tdf);
K=K+asst(K10,id10,tdf)

P=[0;-400;0;-400;0;-150;0;-150]  %-- beban luar

U=solv(K,P)                     %-- perpindahan global

u1=disst(U,id1,T1)              %-- deformasi batang
S1=stret(k1,u1)                 %-- gaya batang
```

```
u2=disst(U,id2,T2)
S2=stret(k2,u2)

u3=disst(U,id3,T3)
S3=stret(k3,u3)

u4=disst(U,id4,T4)
S4=stret(k4,u4)

u5=disst(U,id5,T5)
S5=stret(k5,u5)

u6=disst(U,id6,T6)
S6=stret(k6,u6)

u7=disst(U,id7,T7)
S7=stret(k7,u7)

u8=disst(U,id8,T8)
S8=stret(k8,u8)

u9=disst(U,id9,T9)
S9=stret(k9,u9)

u10=disst(U,id10,T10)
S10=stret(k10,u10)
```

Jika program dipanggil dari *command window* sebagai berikut ini.

```
--> c3_tr
```

Setelah tombol enter ditekan diperoleh hasil sebagai berikut ini.

```
L1 =
 4000
T1 =
 1 0 0 0
 0 1 0 0
 0 0 1 0
 0 0 0 1
k1 =
  369.6000         0 -369.6000         0
         0         0         0         0
 -369.6000         0  369.6000         0
         0         0         0         0
K1 =
  369.6000         0 -369.6000         0
         0         0         0         0
 -369.6000         0  369.6000         0
         0         0         0         0
```

```
L2 =
 4000
T2 =
 1 0 0 0
 0 1 0 0
 0 0 1 0
 0 0 0 1
k2 =
  369.6000         0 -369.6000         0
         0         0         0         0
 -369.6000         0  369.6000         0
         0         0         0         0
K2 =
  369.6000         0 -369.6000         0
         0         0         0         0
 -369.6000         0  369.6000         0
         0         0         0         0
L3 =
 4000
T3 =
 1 0 0 0
 0 1 0 0
 0 0 1 0
 0 0 0 1
k3 =
  369.6000         0 -369.6000         0
         0         0         0         0
 -369.6000         0  369.6000         0
         0         0         0         0
K3 =
  369.6000         0 -369.6000         0
         0         0         0         0
 -369.6000         0  369.6000         0
         0         0         0         0
L4 =
 4000
T4 =
 1 0 0 0
 0 1 0 0
 0 0 1 0
 0 0 0 1
k4 =
  369.6000         0 -369.6000         0
         0         0         0         0
 -369.6000         0  369.6000         0
         0         0         0         0
K4 =
  369.6000         0 -369.6000         0
         0         0         0         0
 -369.6000         0  369.6000         0
         0         0         0         0
```

```
L5 =
 3000
T5 =
  0  1  0  0
 -1  0  0  0
  0  0  0  1
  0  0 -1  0
k5 =
  492.8000         0 -492.8000         0
         0         0         0         0
 -492.8000         0  492.8000         0
         0         0         0         0
K5 =
         0         0         0         0
         0  492.8000         0 -492.8000
         0         0         0         0
         0 -492.8000         0  492.8000
L6 =
 3000
T6 =
  0  1  0  0
 -1  0  0  0
  0  0  0  1
  0  0 -1  0
k6 =
  492.8000         0 -492.8000         0
         0         0         0         0
 -492.8000         0  492.8000         0
         0         0         0         0
K6 =
         0         0         0         0
         0  492.8000         0 -492.8000
         0         0         0         0
         0 -492.8000         0  492.8000
L7 =
 5000
T7 =
    0.8000    0.6000         0         0
   -0.6000    0.8000         0         0
         0         0    0.8000    0.6000
         0         0   -0.6000    0.8000
k7 =
  295.6800         0 -295.6800         0
         0         0         0         0
 -295.6800         0  295.6800         0
         0         0         0         0
K7 =
  189.2352  141.9264 -189.2352 -141.9264
  141.9264  106.4448 -141.9264 -106.4448
 -189.2352 -141.9264  189.2352  141.9264
 -141.9264 -106.4448  141.9264  106.4448
```

```
L8 =
 5000
T8 =
    0.8000   -0.6000         0         0
    0.6000    0.8000         0         0
         0         0    0.8000   -0.6000
         0         0    0.6000    0.8000
k8 =
  295.6800         0 -295.6800         0
         0         0         0         0
 -295.6800         0  295.6800         0
         0         0         0         0
K8 =
  189.2352 -141.9264 -189.2352  141.9264
 -141.9264  106.4448  141.9264 -106.4448
 -189.2352  141.9264  189.2352 -141.9264
  141.9264 -106.4448 -141.9264  106.4448
L9 =
 5000
T9 =
    0.8000    0.6000         0         0
   -0.6000    0.8000         0         0
         0         0    0.8000    0.6000
         0         0   -0.6000    0.8000
k9 =
  295.6800         0 -295.6800         0
         0         0         0         0
 -295.6800         0  295.6800         0
         0         0         0         0
K9 =
  189.2352  141.9264 -189.2352 -141.9264
  141.9264  106.4448 -141.9264 -106.4448
 -189.2352 -141.9264  189.2352  141.9264
 -141.9264 -106.4448  141.9264  106.4448
L10 =
 5000
T10 =
    0.8000   -0.6000         0         0
    0.6000    0.8000         0         0
         0         0    0.8000   -0.6000
         0         0    0.6000    0.8000
k10 =
  295.6800         0 -295.6800         0
         0         0         0         0
 -295.6800         0  295.6800         0
         0         0         0         0
K10 =
  189.2352 -141.9264 -189.2352  141.9264
 -141.9264  106.4448  141.9264 -106.4448
 -189.2352  141.9264  189.2352 -141.9264
  141.9264 -106.4448 -141.9264  106.4448
```

```
tdf =
 8
K =
 Columns 1 to 6
  928.4352  141.9264 -369.6000         0         0         0
  141.9264  599.2448         0         0         0 -492.8000
 -369.6000         0  928.4352 -141.9264 -189.2352  141.9264
         0         0 -141.9264  599.2448  141.9264 -106.4448
         0         0 -189.2352  141.9264  748.0704         0
         0 -492.8000  141.9264 -106.4448         0  705.6896
 -189.2352 -141.9264         0         0 -369.6000         0
 -141.9264 -106.4448         0 -492.8000         0         0
 Columns 7 to 8
 -189.2352 -141.9264
 -141.9264 -106.4448
         0         0
         0 -492.8000
 -369.6000         0
         0         0
  748.0704         0
         0  705.6896
P =
    0
 -400
    0
 -400
    0
 -150
    0
 -150
U =
   -0.0398
   -7.2890
    0.0398
   -7.2890
    0.9323
   -6.4101
   -0.9323
   -6.4101
u1 =
         0
         0
   -0.0398
   -7.2890

S1 =
   14.7186
         0
  -14.7186
         0
```

```
u2 =
   -0.0398
   -7.2890
    0.0398
   -7.2890
S2 =
  -29.4372
         0
   29.4372
         0
u3 =
    0.0398
   -7.2890
         0
         0
S3 =
   14.7186
         0
  -14.7186
         0
u4 =
    0.9323
   -6.4101
   -0.9323
   -6.4101
S4 =
  689.1775
         0
 -689.1775
         0
u5 =
   -7.2890
    0.0398
   -6.4101
   -0.9323
S5 =
 -433.1169
         0
  433.1169
         0
u6 =
   -7.2890
   -0.0398
   -6.4101
    0.9323
S6 =
 -433.1169
         0
  433.1169
         0
```

```
u7 =
         0
         0
   -3.1002
   -5.6875
S7 =
  916.6667
         0
 -916.6667
         0
u8 =
    4.5919
   -4.5687
    4.4053
   -5.8073
S8 =
   55.1948
         0
  -55.1948
         0
u9 =
   -4.4053
   -5.8073
   -4.5919
   -4.5687
S9 =
   55.1948
         0
  -55.1948
         0
u10 =
    3.1002
   -5.6875
         0
         0


S10 =
  916.6667
         0
 -916.6667
         0
-->
```

Dari keluaran tersebut gaya-gaya batang dapat disusun seperti pada Tabel 4.5.

Tabel 4.5. Gaya-gaya batang contoh 3 struktur rangka bidang

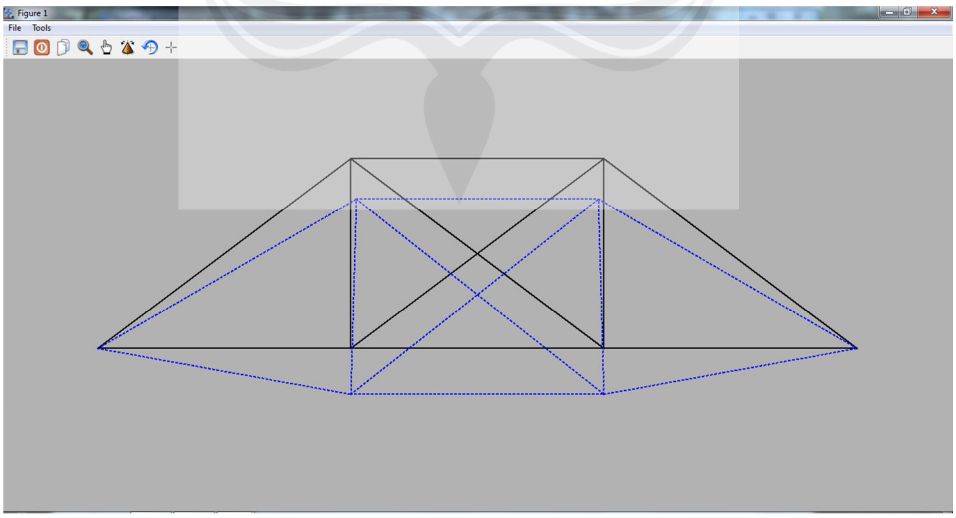
Batang	Gaya (kN)
1	-14.7186
2	29.4372
3	-14.7186
4	-689.1775
5	433.1169
6	433.1169
7	-916.6667
8	-55.1948
9	-55.1948
10	-916.6667

Untuk menampilkan gambar struktur sebelum dan sesudah dibebani dapat dilakukan dengan menjalankan perintah berikut ini.

```
M=[n1;n2;    % batang 1
n2;n3;       % batang 2
n3;n4;       % batang 3
n5;n6;       % batang 4
n2;n5;       % batang 5
n3;n6;       % batang 6
n1;n5;       % batang 7
n5;n3;       % batang 8
n2;n6;       % batang 9
n6;n4]       % batang 10

plot2d(M,'-k','LineWidth',2)  % plot 'undeformed'


Un=[0; 0;          % node 1
U(1); U(2)         % node 2
U(3); U(4)         % node 3
0; 0;              % node 4
U(5); U(6)         % node 5
U(7); U(8)]        % node 6

sc=100;                 % faktor skala
n1n=n1+sc*Un(1:2)'      % lokasi baru
n2n=n2+sc*Un(3:4)'
n3n=n3+sc*Un(5:6)'
```

```
n4n=n4+sc*Un(7:8)'
n5n=n5+sc*Un(9:10)'
n6n=n6+sc*Un(11:12)'

Mn=[n1n;n2n;           % batang 1 -- deformed mesh
n2n;n3n;               % batang 2
n3n;n4n;               % batang 3
n5n;n6n;               % batang 4
n2n;n5n;               % batang 5
n3n;n6n;               % batang 6
n1n;n5n;               % batang 7
n5n;n3n;               % batang 8
n2n;n6n;               % batang 9
n6n;n4n]               % batang 10

plot2d(Mn,':b','LineWidth',2) % plot 'deformed'
```

Perintah tersebut dapat disimpan dalam suatu berkas, misal plot_c3_tr.m, lalu dijalankan pada *command window*. Jika dijalankan akan diperoleh hasil seperti terlihat pada Gambar 4.11.

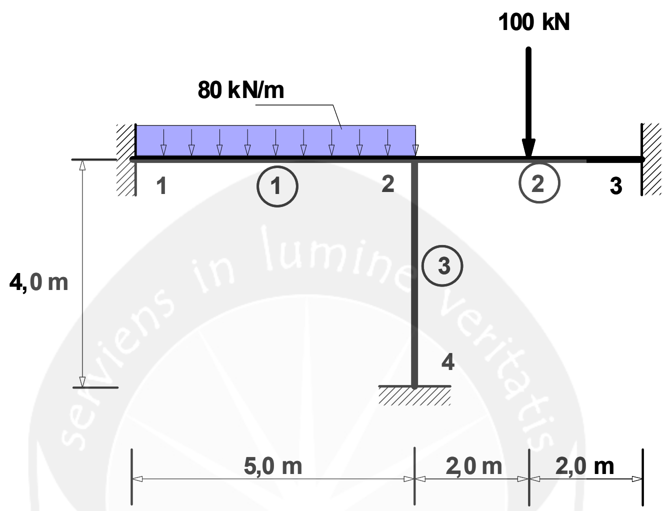


Gambar 4.11. Geometri contoh 3 struktur rangka bidang

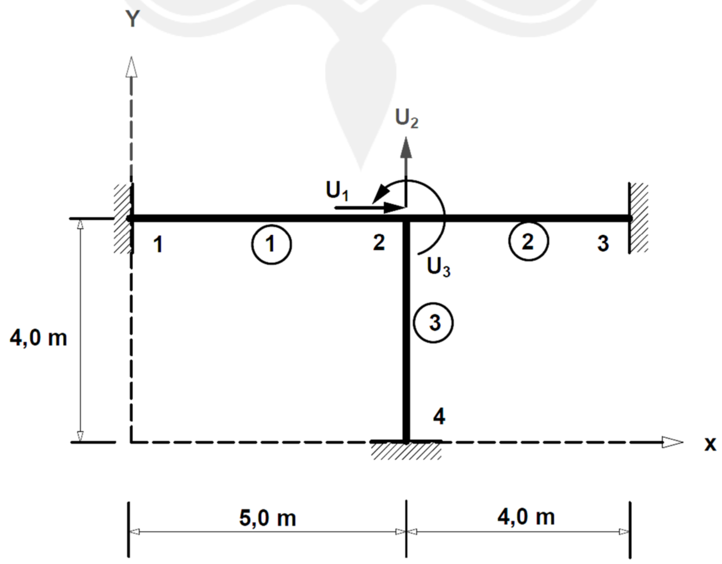
4.2 APLIKASI PADA STRUKTUR PORTAL BIDANG

4.2.1. Contoh 1 portal bidang

Suatu portal seperti terlihat pada Gambar 4.12. Ukuran batang: b = 0,20 m, h = 0,45 m. Modulus elastik bahan: E = 2,52 x 10^7 kN/m^2. Derajat kebebasan struktur = 3, seperti ditunjukkan pada Gambar 4.13. Hubungan antar batang ditentukan seperti pada Tabel 4.6.



Gambar 4.12. Contoh 1 portal bidang



Gambar 4.13. Derajat kebebasan contoh 1 portal bidang

Tabel 4.6. Hubungan antar batang contoh 1 portal bidang

Batang	Titik-i	Titik-j
1	1	2
2	2	3
3	4	2

Penyelesaian dengan dengan FreeMat sebagai berikut ini. Program disimpan dalam berkas c1_fr.m

```
% contoh 1 portal bidang
n1 =coor(0,4);
n2=coor(5,4);
n3=coor(9,4);
n4=coor(5,0);

E=2.52e7;               %Data material

I1=1/12*0.20*0.45^3
A1= 0.20*0.45
[L1,T1]=memf(n1,n2)  %menghitung L dan T
k1=klf(E,A1,I1,L1)   %Matriks kekakuan dalam koordinat lokal
K1=kg(k1,T1)         %Matriks kekakuan dalam koordinat global
ID1=[0 0 0 1 2 3]    %Vektor tujuan

I2=I1
A2=A1
[L2,T2]=memf(n2,n3)
k2=klf(E,A2,I2,L2)
K2=kg(k2,T2)
ID2=[1 2 3 0 0 0]

I3=I1
A3=A1
[L3,T3]=memf(n4,n2)
k3=klf(E,A3,I3,L3)
K3=kg(k3,T3)
ID3=[0 0 0 1 2 3]

dof=3
K=assf(K1,ID1,dof);   %Perakitan K: kontribusi batang 1
K=K+assf(K2,ID2,dof); %              +kontribusi batang 2
K=K+assf(K3,ID3,dof)  %              +kontribusi batang 3

So1=feq(-80,0,5,5)    %Gaya jepit ujung: feq(q,a1,a2,L)
So2=fep(-100,2,4)     % ------''-------: fep(P,a1,L)
So3=zeros(6,1)        %Gaya jepit ujung So=[0] (6x1)

Pe1=peqj(So1,T1,ID1,dof)  %beban titik ekivalen peqj dari batang 1
```

```
Pe2=peqj(So2,T2,ID2,dof)
P=Pe1+Pe2
```

```
U=solv(K,P)         %Perpindahan

u1=dissf(U,ID1,T1)  %Deformasi batang 1
S1=stref(k1,u1,So1) %Gaya batang 1

u2=dissf(U,ID2,T2)
S2=stref(k2,u2,So2)

u3=dissf(U,ID3,T3)
S3=stref(k3,u3,So3)
```

Jika program dijalankan dihasilkan sebagai berikut ini.

```
--> c1_fr
I1 =
    0.0015
A1 =
    0.0900
L1 =
 5
T1 =
 1 0 0 0 0 0
 0 1 0 0 0 0
 0 0 1 0 0 0
 0 0 0 1 0 0
 0 0 0 0 1 0
 0 0 0 0 0 1
k1 =
   1.0e+005 *
    4.5360         0         0   -4.5360         0         0
         0    0.0367    0.0919         0   -0.0367    0.0919
         0    0.0919    0.3062         0   -0.0919    0.1531
   -4.5360         0         0    4.5360         0         0
         0   -0.0367   -0.0919         0    0.0367   -0.0919
         0    0.0919    0.1531         0   -0.0919    0.3062
K1 =
   1.0e+005 *
    4.5360         0         0   -4.5360         0         0
         0    0.0367    0.0919         0   -0.0367    0.0919
         0    0.0919    0.3062         0   -0.0919    0.1531
   -4.5360         0         0    4.5360         0         0
         0   -0.0367   -0.0919         0    0.0367   -0.0919
         0    0.0919    0.1531         0   -0.0919    0.3062
ID1 =
 0 0 0 1 2 3
```

```
I2 =
    0.0015
A2 =
    0.0900
L2 =
 4
T2 =
 1 0 0 0 0 0
 0 1 0 0 0 0
 0 0 1 0 0 0
 0 0 0 1 0 0
 0 0 0 0 1 0
 0 0 0 0 0 1
k2 =
   1.0e+005 *
    5.6700         0         0   -5.6700         0         0
         0    0.0718    0.1435         0   -0.0718    0.1435
         0    0.1435    0.3827         0   -0.1435    0.1914
   -5.6700         0         0    5.6700         0         0
         0   -0.0718   -0.1435         0    0.0718   -0.1435
         0    0.1435    0.1914         0   -0.1435    0.3827



K2 =
   1.0e+005 *
    5.6700         0         0   -5.6700         0         0
         0    0.0718    0.1435         0   -0.0718    0.1435
         0    0.1435    0.3827         0   -0.1435    0.1914
   -5.6700         0         0    5.6700         0         0
         0   -0.0718   -0.1435         0    0.0718   -0.1435
         0    0.1435    0.1914         0   -0.1435    0.3827
ID2 =
 1 2 3 0 0 0
I3 =
    0.0015
A3 =
    0.0900
L3 =
 4
T3 =
  0  1  0  0  0  0
 -1  0  0  0  0  0
  0  0  1  0  0  0
  0  0  0  0  1  0
  0  0  0 -1  0  0
  0  0  0  0  0  1
```

```
k3 =
   1.0e+005 *
    5.6700         0         0   -5.6700         0         0
         0    0.0718    0.1435         0   -0.0718    0.1435
         0    0.1435    0.3827         0   -0.1435    0.1914
   -5.6700         0         0    5.6700         0         0
         0   -0.0718   -0.1435         0    0.0718   -0.1435
         0    0.1435    0.1914         0   -0.1435    0.3827
K3 =
   1.0e+005 *
    0.0718         0   -0.1435   -0.0718         0   -0.1435
         0    5.6700         0         0   -5.6700         0
   -0.1435         0    0.3827    0.1435         0    0.1914
   -0.0718         0    0.1435    0.0718         0    0.1435
         0   -5.6700         0         0    5.6700         0
   -0.1435         0    0.1914    0.1435         0    0.3827
ID3 =
 0 0 0 1 2 3
dof =
 3
K =
   1.0e+006 *
    1.0278         0    0.0144
         0    0.5779    0.0052
    0.0144    0.0052    0.1072
So1 =
         0
  200.0000
  166.6667
         0
  200.0000
 -166.6667
So2 =
   0
  50
  50
   0
  50
 -50
So3 =
 0
 0
 0
 0
 0
 0
```

```
Pe1 =
         0
 -200.0000
  166.6667
Pe2 =
   0
 -50
 -50
P =
         0
 -250.0000
  116.6667
U =
   1.0e-003 *
   -0.0155
   -0.4426
    1.1121
u1 =
   1.0e-003 *
         0
         0
         0
   -0.0155
   -0.4426
    1.1121
S1 =
    7.0443
  211.8412
  187.7571
   -7.0443
  188.1588
 -128.5510
u2 =
   1.0e-003 *
   -0.0155
   -0.4426
    1.1121
         0
         0
         0
S2 =
   -8.8054
   62.7851
   86.2109
    8.8054
   37.2149
  -35.0705
```

```
u3 =
   1.0e-003 *
         0
         0
         0
   -0.4426
    0.0155
    1.1121
S3 =
  250.9439
   15.8497
   21.0586
 -250.9439
  -15.8497
   42.3401
-->
```

Dari keluaran tersebut, gaya-gaya batang adalah seperti pada Tabel 4.7.

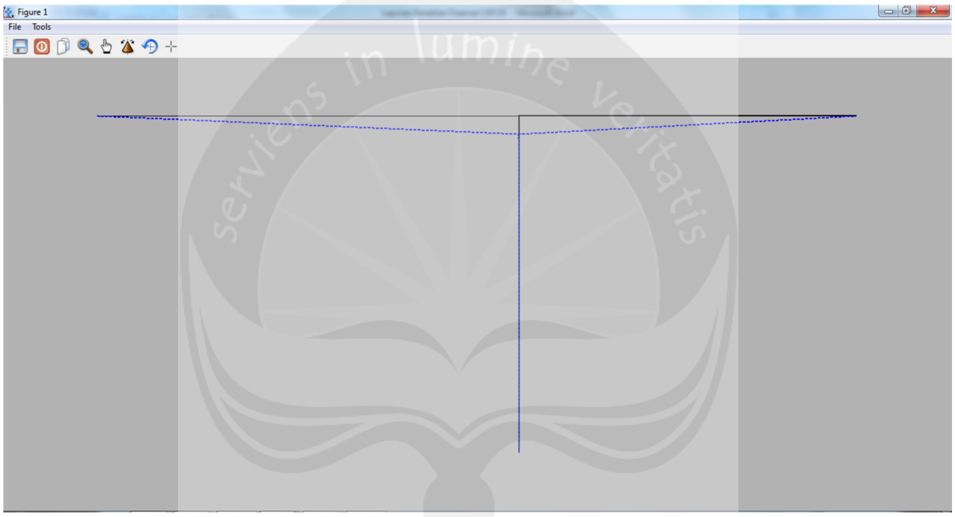
Tabel 4.7 Gaya batang contoh 1 portal bidang

Batang	Titik simpul	Gaya normal (kN)	Gaya geser (kN)	Momen (kNm)
1	1	7,0443	211,8412	187,7571
	2	-7,0443	188,1588	-128,5510
2	2	-8,8054	62,7851	86,2109
	3	8,8054	37,2149	-35,0705
3	4	250,9439	15,8497	21,0586
	2	-250,9439	-15,8497	42,3401

Jika geometri struktur ingin digambarkan, program berikut ini dapat digunakan. Jika dijalankan, hasilnya dapat dilihat pada Gambar 4.14.

```
M=[n1;n2;   % batang 1
n2;n3;      % batang 2
n4;n2]      % batang 3

plot2d(M,'-k','LineWidth',2) % plot 'undeformed'


Un=[0; 0; 0;            % node 1
U(1); U(2); U(3);       % node 2
0; 0; 0;                % node 3
0; 0; 0]                % node 4
```

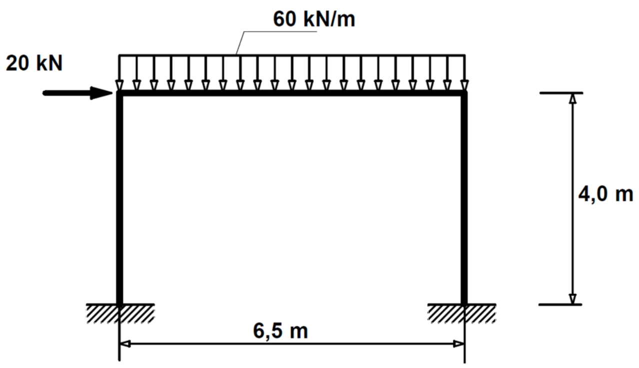
```
sc=500;                 %faktor skala
n1n=n1+sc*Un(1:2)'      %lokasi baru
n2n=n2+sc*Un(4:5)'
n3n=n3+sc*Un(7:8)'
n4n=n4+sc*Un(10:11)'


Mn=[n1n;n2n;            % batang 1
n2n;n3n;                % batang 2
n4n;n2n]                % batang 3

plot2d(Mn,':b','LineWidth',2)      %plot 'deformed'
```



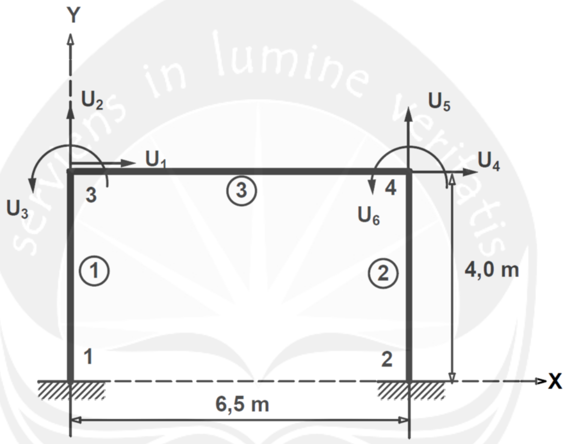
Gambar 4.14. Struktur contoh 1 portal bidang, sebelum dan sesuadah dibebani

4.2.2. Contoh 2 portal bidang

Suatu portal seperti terlihat pada Gambar 4.15. Ukuran balok: 0,25 × 0,50 (m), kolom: 0,40 ×0,40 (m). Modulus elastik E = 2,35 × 10^7 kN/m^2, angka Poisson υ = 0,20, dan $G = \frac{E}{2(1+\nu)}$. Derajat kebebasan = 6 dan ditunjukkan pada Gambar 4.16. Hubungan antar batang seperti pada Tabel 4.16.



Gambar 4.15. Struktur dan beban contoh 2 portal bidang



Gambar 4.16. Derajat kebebasan contph 2 portal bidang

Tabel 4.16. Hubungan antar batang contoh 1 portal bidang

Batang	Ujung-i	Ujung-j
1	1	3
2	2	4
3	3	4

Penyelesaian:

Dalam contoh ini deformasi geser ikut diperhitungkan dengan f_s = 1,2 untuk tampang empat persegi panjang. Program dapat ditulis sebagai berikut ini, dan disimpan dengan nama c2_fr.m

```
clear all
n1=coor(0,0);
n2=coor(6.5,0);
n3=coor(0,4);
n4=coor(6.5,4);

E=2.35e7;                   % modulus elastik
v=0.2;                      % angka Poisson

A1=0.4*0.4;
I1=1/12*0.4*0.4^3;
f1=6/5;
[L1,T1]=memf(n1,n3)
k1=klfs(E,A1,I1,L1,f1,v)    % matriks kekakuan dengan deformasi
geser
K1=kg(k1,T1)
ID1=[0 0 0 1 2 3]

A2=A1;
I2=I1;
f2=f1;
[L2,T2]=memf(n2,n4)
k2=klfs(E,A2,I2,L2,f2,v)
K2=kg(k2,T2)
ID2=[0 0 0 4 5 6]

A3=0.25*0.5;
I3=1/12*0.2*0.45^3;
f3=f1;
[L3,T3]=memf(n3,n4)
k3=klfs(E,A3,I3,L3,f3,v)
K3=kg(k3,T3)
ID3=[1 2 3 4 5 6]

dof=6

K=assf(K1,ID1,dof);           % penggabungan--kontribusi batang 1
K=K+assf(K2,ID2,dof);
K=K+assf(K3,ID3,dof)          % K struktur

So1=zeros(6,1)                % gaya jepit
So2=zeros(6,1)
So3=feq(-60,0,6.5,6.5)

Po3=peqj(So3,T3,ID3,dof)     % beban titik ekivalen
Pj=[20;0;0;0;0;0]            % beban pada titik simpul
P=Po3+Pj

U=solv(K,P)
```

```
u1=dissf(U,ID1,T1)
S1=stref(k1,u1,So1)

u2=dissf(U,ID2,T2)
S2=stref(k2,u2,So2)

u3=dissf(U,ID3,T3)
S3=stref(k3,u3,So3)

disp('    S1        S2        S3')
S=[S1 S2 S3]
```

Jika dijalankan akan menghasilkan sebagai berikut ini.

```
--> c2_fr
L1 =
 4
T1 =
  0  1  0  0  0  0
 -1  0  0  0  0  0
  0  0  1  0  0  0
  0  0  0  0  1  0
  0  0  0 -1  0  0
  0  0  0  0  0  1
k1 =
   1.0e+005 *
    9.4000         0         0   -9.4000         0         0
         0    0.0914    0.1827         0   -0.0914    0.1827
         0    0.1827    0.4908         0   -0.1827    0.2401
   -9.4000         0         0    9.4000         0         0
         0   -0.0914   -0.1827         0    0.0914   -0.1827
         0    0.1827    0.2401         0   -0.1827    0.4908
K1 =
   1.0e+005 *
    0.0914         0   -0.1827   -0.0914         0   -0.1827
         0    9.4000         0         0   -9.4000         0
   -0.1827         0    0.4908    0.1827         0    0.2401
   -0.0914         0    0.1827    0.0914         0    0.1827
         0   -9.4000         0         0    9.4000         0
   -0.1827         0    0.2401    0.1827         0    0.4908
ID1 =
 0 0 0 1 2 3
L2 =
 4
T2 =
  0  1  0  0  0  0
 -1  0  0  0  0  0
  0  0  1  0  0  0
  0  0  0  0  1  0
  0  0  0 -1  0  0
  0  0  0  0  0  1
```

```
k2 =
   1.0e+005 *
    9.4000         0         0   -9.4000         0         0
         0    0.0914    0.1827         0   -0.0914    0.1827
         0    0.1827    0.4908         0   -0.1827    0.2401
   -9.4000         0         0    9.4000         0         0
         0   -0.0914   -0.1827         0    0.0914   -0.1827
         0    0.1827    0.2401         0   -0.1827    0.4908
K2 =
   1.0e+005 *
    0.0914         0   -0.1827   -0.0914         0   -0.1827
         0    9.4000         0         0   -9.4000         0
   -0.1827         0    0.4908    0.1827         0    0.2401
   -0.0914         0    0.1827    0.0914         0    0.1827
         0   -9.4000         0         0    9.4000         0
   -0.1827         0    0.2401    0.1827         0    0.4908
ID2 =
 0 0 0 4 5 6
L3 =
    6.5000
T3 =
 1 0 0 0 0 0
 0 1 0 0 0 0
 0 0 1 0 0 0
 0 0 0 1 0 0
 0 0 0 0 1 0
 0 0 0 0 0 1
k3 =
   1.0e+005 *
    4.5192         0         0   -4.5192         0         0
         0    0.0154    0.0502         0   -0.0154    0.0502
         0    0.0502    0.2180         0   -0.0502    0.1082
   -4.5192         0         0    4.5192         0         0
         0   -0.0154   -0.0502         0    0.0154   -0.0502
         0    0.0502    0.1082         0   -0.0502    0.2180
K3 =
   1.0e+005 *
    4.5192         0         0   -4.5192         0         0
         0    0.0154    0.0502         0   -0.0154    0.0502
         0    0.0502    0.2180         0   -0.0502    0.1082
   -4.5192         0         0    4.5192         0         0
         0   -0.0154   -0.0502         0    0.0154   -0.0502
         0    0.0502    0.1082         0   -0.0502    0.2180
ID3 =
 1 2 3 4 5 6
dof =
 6
```

```
K =
   1.0e+005 *
    4.6106         0    0.1827   -4.5192         0         0
         0    9.4154    0.0502         0   -0.0154    0.0502
    0.1827    0.0502    0.7088         0   -0.0502    0.1082
   -4.5192         0         0    4.6106         0    0.1827
         0   -0.0154   -0.0502         0    9.4154   -0.0502
         0    0.0502    0.1082    0.1827   -0.0502    0.7088
So1 =
 0
 0
 0
 0
 0
 0
So2 =
 0
 0
 0
 0
 0
 0
So3 =
         0
  195.0000
  211.2500
         0
  195.0000
 -211.2500
Po3 =
         0
 -195.0000
 -211.2500
         0
 -195.0000
  211.2500
Pj =
 20
  0
  0
  0
  0
  0
P =
   20.0000
 -195.0000
 -211.2500
         0
 -195.0000
  211.2500
```

```
U =
   1.0e-003 *
    2.0643
   -0.2027
   -3.9860
    1.9006
   -0.2122
    3.0981
u1 =
   1.0e-003 *
         0
         0
         0
   -0.2027
   -2.0643
   -3.9860
S1 =
  190.5583
  -53.9790
  -57.9995
 -190.5583
   53.9790
 -157.9164
u2 =
   1.0e-003 *
         0
         0
         0
   -0.2122
   -1.9006
    3.0981
S2 =
  199.4417
   73.9790
  109.1285
 -199.4417
  -73.9790
  186.7874
u3 =
   1.0e-003 *
    2.0643
   -0.2027
   -3.9860
    1.9006
   -0.2122
    3.0981
```

```
S3 =
   73.9790
  190.5583
  157.9164
  -73.9790
  199.4417
 -186.7874
    S1        S2        S3
S =
  190.5583  199.4417   73.9790
  -53.9790   73.9790  190.5583
  -57.9995  109.1285  157.9164
 -190.5583 -199.4417  -73.9790
   53.9790  -73.9790  199.4417
 -157.9164  186.7874 -186.7874
-->
```

Tabel 4.17. Hasil analisis contoh 2 portal bidang

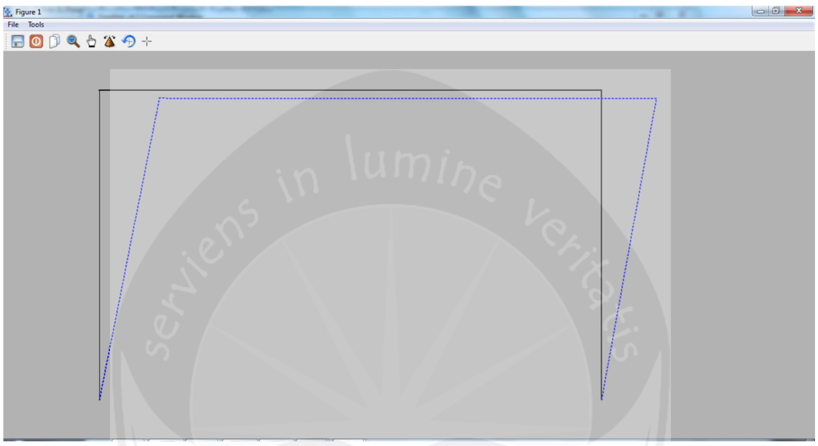
Batang	Titik simpul	Gaya normal (kN)	Gaya geser (kN)	Momen (kNm)
1	1	190.5583	-53.9790	-57.9995
	3	-190.5583	53.9790	-157.9164
2	2	199.4417	73.9790	109.1285
	4	-199.4417	-73.9790	186.7874
3	3	73.9790	190.5583	157.9164
	4	-73.9790	199.4417	-186.7874

Jika geometri struktur ingin digambarkan dapat dilakukan dengan program berikut ini.

```
M=[n1;n3;  % batang 1
n2;n4;     % batang 2
n3;n4]     % batang 3

plot2d(M,'-k','LineWidth',2)   % plot 'undeformed'


Un=[0;    0;    0;            % node 1
    0;    0;    0;            % node 2
   U(1); U(2); U(3)           % node 3
   U(4); U(5); U(6)]          % node 4


sc=500;                    %faktor skala
n1n=n1+sc*Un(1:2)'         % lokasi baru
n2n=n2+sc*Un(4:5)'
n3n=n3+sc*Un(7:8)'
n4n=n4+sc*Un(10:11)'
```

```
Mn=[n1n;n3n;  % batang 1
n2n;n4n;      % batang 2
n3n;n4n]      % batang 3

plot2d(Mn,':b','LineWidth',2)   %plot 'deformed'
```

Sebagai catatan program penggambaran ini belum memperhitungkan perilaku batang yang terlentur. Batang-batang haya digambarkan berdasarkan letak titik awal dan akhir secara lurus. Hasil penggambaran ditunjukkan pada Gambar 4.17.



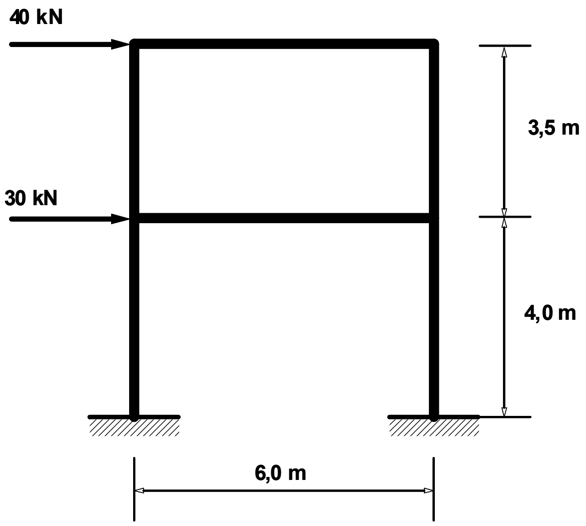
Gambar 4.17. Geometri contoh 2 portal bidang

4.2.3. Contoh 3 portal bidang

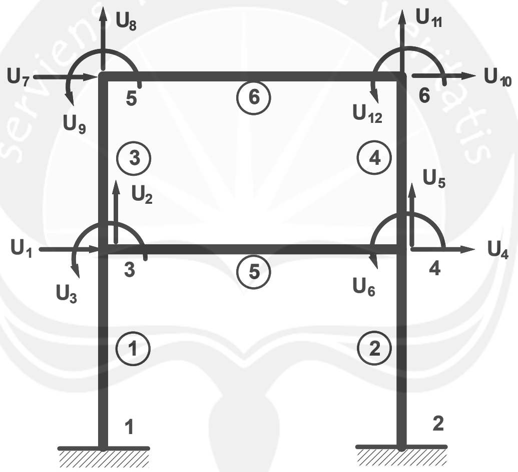
Ditinjau suatu struktur seperti tergambar pada Gambar 4.18 dengan ukuran balok: 0,30×0,45 (m), kolom: 0,40×0,40 (m), E = $2{,}35\times10^{7}$ kN/mm^{2} dan angka Poisson υ = 0,2. Hitung gaya-gaya internal batang dengan memperhitungkan deformasi geser.

Penyelesaian:

Langkah hitungan sama seperti pada contoh-contoh sebelumnya. Derajat kebebasan seperti pada Gambar 4.19. Hubungan antar batang diambil seperti pada Tabel 4.18.



Gambar 4.18 Struktur dan beban contoh 3 portl bidang



Gambar 4.19. Derajat kebebasan dan penomoran contoh 3 portal bidang

Tabel 4.18. Hubungan antar batang

Batang	Ujung-i	Ujung-j
1	1	3
2	2	4
3	3	5
4	4	6
5	3	4
6	5	6

Program REALIN dapat ditulis sebagai berikut ini.

```
clear all

n1=coor(0,0);
n2=coor(6,0);
n3=coor(0,4);
n4=coor(6,4);
n5=coor(0,7.5);
n6=coor(6,7.5);

E=2.35e7;
v=0.2;

A1=0.4*0.4;
I1=1/12*0.4*0.4^3;
f1=6/5;
[L1,T1]=memf(n1,n3);
k1=klfs(E,A1,I1,L1,f1,v);
K1=kg(k1,T1);
ID1=[0 0 0 1 2 3];

A2=A1;
I2=I1;
f2=f1;
[L2,T2]=memf(n2,n4);
k2=klfs(E,A2,I2,L2,f2,v);
K2=kg(k2,T2);
ID2=[0 0 0 4 5 6];

A3=A1;
I3=I1;
f3=f1;
[L3,T3]=memf(n3,n5);
k3=klfs(E,A3,I3,L3,f3,v);
K3=kg(k3,T3);
ID3=[1 2 3 7 8 9];

A4=A1;
I4=I1;
f4=f1;
[L4,T4]=memf(n4,n6);
k4=klfs(E,A4,I4,L4,f4,v);
K4=kg(k4,T4);
ID4=[4 5 6 10 11 12];

A5=0.30*0.45;
I5=1/12*0.30*0.45^3;
f5=6/5;
[L5,T5]=memf(n3,n4);
k5=klfs(E,A5,I5,L5,f5,v);
K5=kg(k5,T5);
```

```
ID5=[1 2 3 4 5 6];

A6=A5;
I6=I5;
f6=f5;
[L6,T6]=memf(n5,n6);
k6=klfs(E,A6,I6,L6,f6,v);
K6=kg(k6,T6);
ID6=[7 8 9 10 11 12];


dof=12;

K=assf(K1,ID1,dof);
K=K+assf(K2,ID2,dof);
K=K+assf(K3,ID3,dof);
K=K+assf(K4,ID4,dof);
K=K+assf(K5,ID5,dof);
K=K+assf(K6,ID6,dof);

So1=zeros(6,1);
So2=zeros(6,1);
So3=zeros(6,1);
So4=zeros(6,1);
So5=zeros(6,1);
So6=zeros(6,1);

P=[30;0;0;0;0;0;40;0;0;0;0;0]

U=solv(K,P)

u1=dissf(U,ID1,T1)
S1=stref(k1,u1,So1)

u2=dissf(U,ID2,T2)
S2=stref(k2,u2,So2)

u3=dissf(U,ID3,T3)
S3=stref(k3,u3,So3)

u4=dissf(U,ID4,T4)
S4=stref(k4,u4,So4)

u5=dissf(U,ID5,T5)
S5=stref(k5,u5,So5)

u6=dissf(U,ID6,T6)
S6=stref(k6,u6,So6)
```

Jika program dijalankan diperoleh hasil sebagai berikut ini.

```
--> c3_fr
P =
 30
  0
  0
  0
  0
  0
 40
  0
  0
  0
  0
  0
U =
   1.0e-002 *
    0.6812
    0.0043
   -0.1486
    0.6783
   -0.0043
   -0.1481
    1.2387
    0.0057
   -0.0855
    1.2349
   -0.0057
   -0.0854
u1 =
   1.0e-003 *
         0
         0
         0
    0.0431
   -6.8116
   -1.4864
S1 =
  -40.4691
   35.0753
   88.7799
   40.4691
  -35.0753
   51.5214
u2 =
   1.0e-003 *
         0
         0
         0
   -0.0431
   -6.7834
   -1.4805
```

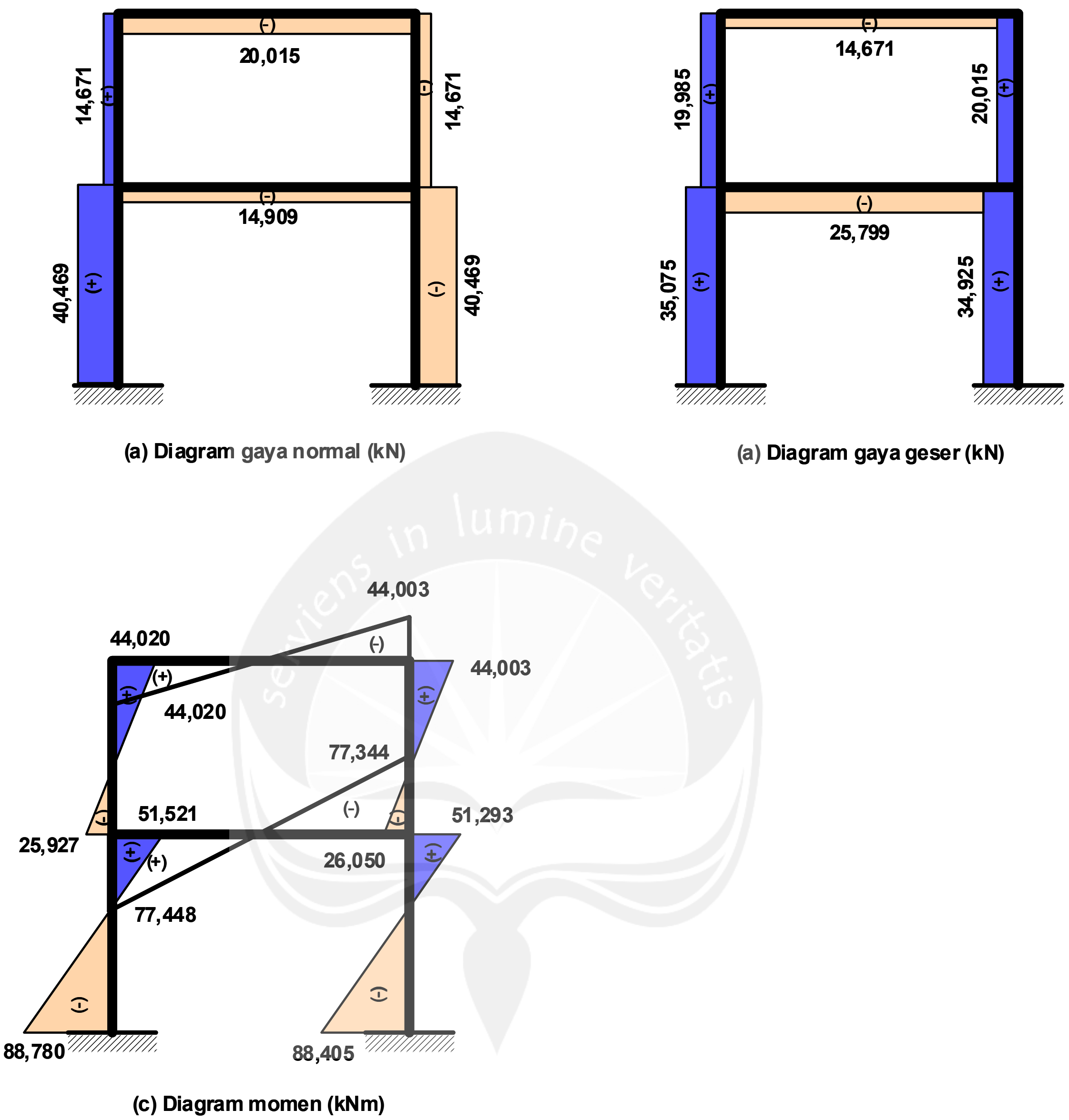
```
S2 =
   40.4691
   34.9247
   88.4052
  -40.4691
  -34.9247
   51.2934
u3 =
   1.0e-002 *
    0.0043
   -0.6812
   -0.1486
    0.0057
   -1.2387
   -0.0855
S3 =
  -14.6705
   19.9848
   25.9266
   14.6705
  -19.9848
   44.0201
u4 =
   1.0e-002 *
   -0.0043
   -0.6783
   -0.1481
   -0.0057
   -1.2349
   -0.0854
S4 =
   14.6705
   20.0152
   26.0501
  -14.6705
  -20.0152
   44.0031
u5 =
   1.0e-003 *
    6.8116
    0.0431
   -1.4864
    6.7834
   -0.0431
   -1.4805
```

```
S5 =
   14.9094
  -25.7986
  -77.4480
  -14.9094
   25.7986
  -77.3436
u6 =
   1.0e-002 *
    1.2387
    0.0057
   -0.0855
    1.2349
   -0.0057
   -0.0854
S6 =
   20.0152
  -14.6705
  -44.0201
  -20.0152
   14.6705
  -44.0031
-->
```

Hasil analisis ditunjukkan pada Tabel 4.19 dan Gambar 4.20. Hasil analisis menunjukkan kesesuaian yang baik antara analisis pada contoh ini dengan hasil analisis program ETABS Nonlinear. Perbedaan pada tanda hanya menunjukkan perbedaan alam pengambilan perjanjian tanda.

Tabel 4.19. Hasil analisis contoh 3 portal bidang

Batang	Titik simpul	Gaya normal (kN)		Gaya geser (kN)		Momen (kNm)	
		Analisis	ETABS	Analisis	ETABS	Analisis	ETABS
1	1	-40,469	40,469	35,075	35,075	88,780	88,780
	3	40,469	40,469	-35,075	35,075	51,521	-51,521
2	2	40,469	-40,469	34,925	34,925	88,405	88,405
	4	-40,469	-40,469	-34,925	34,925	51,293	-51,293
3	3	-14,671	14.671	19,985	19.985	25,927	25.927
	5	14,671	14.671	-19,985	19.985	44,020	-44.020
4	4	14,671	-14,671	20,015	20.015	26,050	26,050
	6	-14,671	-14,671	-20,015	20,015	44,003	-44,003
5	3	14,909	-14,909	-25,799	25,799	-77,448	77,448
	4	-14,909	-14,909	25,799	25,799	-77,344	-77,344
6	5	20,015	-20,015	-14,671	14,671	-44,020	44,020
	6	-20,015	-20,015	14,671	14,671	-44,003	-44,003



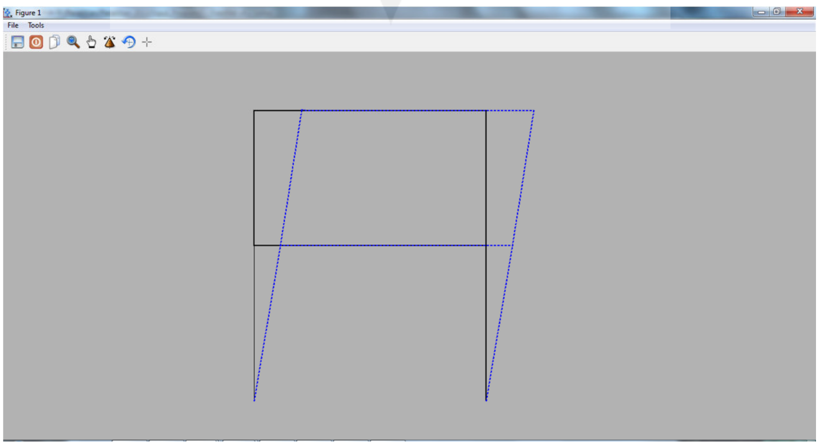
Gambar 4.20. Diagram gaya-internal contoh 3 portal bidang.

Jika geometri struktur ingin digambar, dapat dilakukan dengan program berikut ini.

```
M=[n1;n3;    % batang 1
n2;n4;       % batang 2
n3;n5;       % batang 3
n4;n6;       % batang 4
n3;n4        % batang 5
n5;n6];      % batang 6
```

```
plot2d(M,'-k','LineWidth',2)   % plot 'undeformed'


Un=[0;     0;     0;            % node 1
    0;     0;     0;            % node 2
   U(1);  U(2);  U(3)           % node 3
   U(4);  U(5);  U(6);          % node 4
   U(7);  U(8);  U(9);          % node 5
   U(10); U(11); U(12)];        % node 6


sc=100;                  %faktor skala
n1n=n1+sc*Un(1:2)';      % lokasi baru
n2n=n2+sc*Un(4:5)';
n3n=n3+sc*Un(7:8)';
n4n=n4+sc*Un(10:11)';
n5n=n5+sc*Un(13:14)';
n6n=n6+sc*Un(16:17)';


Mn=[n1n;n3n;  % batang 1
n2n;n4n;      % batang 2
n3n;n5n;      % batang 3
n4n;n6n;      % batang 4
n3n;n4n       % batang 5
n5n;n6n];     % batang 6

plot2d(Mn,':b','LineWidth',2)   %plot 'deformed'
```

Jika program dijalankan, akan dihasilkan tampilan seperti pada Gambar 4.21.



Gambar 4.21. Geometri struktur contoh 3 portal bidang

4.2.4. Contoh 4 portal bidang

Dalam contoh ini, diambil struktur yang sama dengan contoh 3 tetapi dengan memperhitungkan daerah kaku ujung-ujung batang. Daerah kaku diambil seperti pada Tabel 4.20.

Tabel 4.20. Daerah kaku ujung batang contoh 4 portal bidang

Batang	Daerah kaku (m)	
	Ujung -i	Ujung-j
1	0	0,45
2	0	0,45
3	0	0,45
4	0	0,45
5	0,2	0,2
6	0,2	0,2

Program REALIN selanjutnya dapat ditulis sebagai berikut ini.

```
clear all

n1=coor(0,0);
n2=coor(6,0);
n3=coor(0,4);
n4=coor(6,4);
n5=coor(0,7.5);
n6=coor(6,7.5);

E=2.35e7;
v=0.2;

A1=0.4*0.4;
I1=1/12*0.4*0.4^3;
f1=6/5;
[L1,T1]=memf(n1,n3);
ri1=0; rj1=0.45;     % ri, rj = panjang daerah kaku ujung-i dan -j
[k1,k1r,T1r]=klfg(E,A1,I1,L1,f1,v,ri1,rj1);  % k1=kekakuan bag. dalam
K1=kg(k1r,T1);                              % k1r=kekakuan pada sumbu
ID1=[0 0 0 1 2 3];

A2=A1;
I2=I1;
f2=f1;
[L2,T2]=memf(n2,n4);
[k2,k2r,T2r]=klfg(E,A2,I2,L2,f2,v,ri2,rj2);
```

```
K2=kg(k2r,T2);
ID2=[0 0 0 4 5 6];

A3=A1;
I3=I1;
f3=f1;
[L3,T3]=memf(n3,n5);
ri3=0; rj3=0.45;
[k3,k3r,T3r]=klfg(E,A3,I3,L3,f3,v,ri3,rj3);
K3=kg(k3r,T3);
ID3=[1 2 3 7 8 9];

A4=A1;
I4=I1;
f4=f1;
[L4,T4]=memf(n4,n6);
ri4=0; rj4=0.45;
[k4,k4r,T4r]=klfg(E,A4,I4,L4,f4,v,ri4,rj4);
K4=kg(k4r,T4);
ID4=[4 5 6 10 11 12];

A5=0.30*0.45;
I5=1/12*0.30*0.45^3;
f5=6/5;
[L5,T5]=memf(n3,n4);
ri5=0.20; rj5=0.20;
[k5,k5r,T5r]=klfg(E,A5,I5,L5,f5,v,ri5,rj5);
K5=kg(k5r,T5);
ID5=[1 2 3 4 5 6];

A6=A5;
I6=I5;
f6=f5;
[L6,T6]=memf(n5,n6);
ri6=0.20; rj6=0.20;
[k6,k6r,T6r]=klfg(E,A6,I6,L6,f6,v,ri6,rj6);
K6=kg(k6r,T6);
ID6=[7 8 9 10 11 12];


dof=12;

K=assf(K1,ID1,dof);
K=K+assf(K2,ID2,dof);
K=K+assf(K3,ID3,dof);
K=K+assf(K4,ID4,dof);
K=K+assf(K5,ID5,dof);
K=K+assf(K6,ID6,dof);

So1=zeros(6,1);
So2=zeros(6,1);
```

```
So3=zeros(6,1);
So4=zeros(6,1);
So5=zeros(6,1);
So6=zeros(6,1);

P=[30;0;0;0;0;0;40;0;0;0;0;0]

U=solv(K,P)

u1r=dissf(U,ID1,T1)   %% u1r = deformasi pada sumbu
u1=T1r*u1r            %% u1  = deformasi pada bagian dalam batang
S1=stref(k1,u1,So1)

u2r=dissf(U,ID2,T2)    %%
u2=T2r*u2r             %%
S2=stref(k2,u2,So2)

u3r=dissf(U,ID3,T3)  %%
u3=T3r*u3r
S3=stref(k3,u3,So3)

u4r=dissf(U,ID4,T4)  %%
u4=T4r*u4r
S4=stref(k4,u4,So4)

u5r=dissf(U,ID5,T5)  %%
u5=T5r*u5r
S5=stref(k5,u5,So5)

u6r=dissf(U,ID6,T6)  %%
u6=T6r*u6r
S6=stref(k6,u6,So6)
```

Jika program dijalankan diperoleh hasil sebagai berikut ini.

```
--> c4_fr
P =
 30
  0
  0
  0
  0
  0
 40
  0
  0
  0
  0
  0
```

```
U =
   1.0e-002 *
    0.5557
    0.0041
   -0.1282
    0.5530
   -0.0041
   -0.1276
    1.0004
    0.0054
   -0.0761
    0.9969
   -0.0054
   -0.0760
u1r =
   1.0e-003 *
          0
          0
          0
    0.0409
   -5.5568
   -1.2821
u1 =
   1.0e-003 *
          0
          0
          0
    0.0409
   -4.9799
   -1.2821
S1 =
  -43.2713
   35.0804
   80.3731
   43.2713
  -35.0804
   44.1624
u2r =
   1.0e-003 *
          0
          0
          0
   -0.0409
   -5.5305
   -1.2758
```

```
u2 =
   1.0e-003 *
         0
         0
         0
   -0.0409
   -4.9564
   -1.2758
S2 =
   43.2713
   34.9196
   79.9994
  -43.2713
  -34.9196
   43.9652
u3r =
   1.0e-002 *
    0.0041
   -0.5557
   -0.1282
    0.0054
   -1.0004
   -0.0761
u3 =
   1.0e-003 *
    0.0409
   -5.5568
   -1.2821
    0.0538
   -9.6616
   -0.7611
S3 =
  -16.0059
   19.9805
   21.9073
   16.0059
  -19.9805
   39.0334
u4r =
   1.0e-003 *
   -0.0409
   -5.5305
   -1.2758
   -0.0538
   -9.9688
   -0.7604
```

```
u4 =
   1.0e-003 *
   -0.0409
   -5.5305
   -1.2758
   -0.0538
   -9.6266
   -0.7604
S4 =
   16.0059
   20.0195
   22.0574
  -16.0059
  -20.0195
   39.0020
u5r =
   1.0e-003 *
    5.5568
    0.0409
   -1.2821
    5.5305
   -0.0409
   -1.2758
u5 =
   1.0e-003 *
    5.5568
   -0.2156
   -1.2821
    5.5305
    0.2143
   -1.2758
S5 =
   14.9001
  -27.2654
  -76.4028
  -14.9001
   27.2654
  -76.2833
u6r =
   1.0e-002 *
    1.0004
    0.0054
   -0.0761
    0.9969
   -0.0054
   -0.0760
```

```
u6 =
   1.0e-002 *
    1.0004
   -0.0098
   -0.0761
    0.9969
    0.0098
   -0.0760
S6 =
   20.0195
  -16.0059
  -44.8234
  -20.0195
   16.0059
  -44.8096
-->
```

Hasil analisis dibandingkan dengan ETABS Nonlinear seperti ditunjukkan pada Tabel 4.21. Perbedaan tanda menenunjukkan perbedaan dalam pengambilan perjanjian tanda. Dari perbandingan hasil analisis tampak bahwa terdapat kesesuaian yang baik antara program REALIN dan ETABS Nonlinear.

Tabel 4.21. Perbandingan hasi contoh portal bidang dengan ETABS

Batang	Titik simpul	Gaya normal (kN)		Gaya geser (kN)		Momen (kNm)	
		Analisis	ETABS	Analisis	ETABS	Analisis	ETABS
1	1	-43.2713	43.264	35.0804	35.086	80.3731	80.408
	3	43.2713	43.264	-35.0804	35.086	44.1624	-44.147
2	2	43.2713	-43.264	34.9196	34.914	79.9994	80.008
	4	-43.2713	-43.264	-34.9196	34.914	43.9652	-43.936
3	3	-16.0059	16.002	19.9805	19.979	21.9073	21.913
	5	16.0059	16.002	-19.9805	19.979	39.0334	-39.024
4	4	16.0059	-16.002	20.0195	20.021	22.0574	22.073
	6	-16.0059	-16.002	-20.0195	20.021	39.0020	-38.990
5	3	14.9001	-14.893	-27.2654	27.262	-76.4028	76.397
	4	-14.9001	-14.893	27.2654	27.262	-76.2833	-76.269
6	5	20.0195	-20.021	-16.0059	16.002	-44.8234	44.814
	6	-20.0195	-20.021	16.0059	16.002	-44.8096	-44.799

BAB V
PENUTUP

Program REALIN telah dikembangkan dalam FreeMat. Karena FreeMat merupakan *open source* maka sangat cocok digunakan terutama di kalangan perguruan tinggi untuk digunakan mahasiswa mau pun staf pengajar. Program REALIN dibuat dengan maksud agar dapat digunakan sebagai bahan pembelajaran untuk memahami metode kekakuan. Beberapa sub program telah dikembangkan untuk memudahkan dalam analisis. Walau pun demikian langkah-langkah untuk penyelesaian dalam REALIN harus ditentukan oleh pengguna.

Program REALIN dapat digunakan untuk melakukan analisis struktur rangka bidang dan portal bidang. Untuk portal bidang pengguna dapat menentukan apakah pengaruh deformasi geser akan diperhitungkan atau tidak diperhitungkan. Selain itu, jika diinginkan, daerah kaku pada ujung-ujung batang juga dapat diperhitungkan. Perbandingan analisis degan ETABS Nonlinear menunjukkan kesesuaian yang baik antara program REALIN dan ETABS Nonlinear.

PUSTAKA

Arfiadi, Y. (1997). "*Pengembangan program bantu simbolik untuk analisis struktur dengan menggunakan Matlab*", Laporan Penelitian, Program Studi Teknik Sipil, Fakultas Teknik, Universitas Atma Jaya Yogyakarta.

Arfiadi, Y. (2003). Program bantu untuk pengajaran dan pemahaman metoda matriks kekakuan bagi mahasiswa teknik, Lokakarya Sekitar Mekanika Rekayasa, Departemen Teknik Sipil, Fakultas Teknik Sipil dan Perencanaan, Institut Teknologi Bandung, 21 Agustus.

Arfiadi, Y. (2011). *Analisis struktur dengan metode matriks kekakuan.* Cahaya Atma Pustaka, Yogyakarta.

Arfiadi, Y and Hadi, MNS (2002). "*Development of Matrix Method Based Structural Analysis Toolbox in Matlab*", Proceedings of The Sixth International Conference on Computational Structures Technology, BHv Topping and Z. Bitmar (editors), Civil Comp Press, Stirling, Scotland.

Austrell, E.E., Dahlblom, O., Lindemann, J., Olsson, A., Olsson, K.G., Persson, K., Petersson, H., Ristinmaa, M., Sandberg, G., dan Wernberg, P.A. (2004). *CALFEM: a finite element toolbox version 3.4*. Division of Structural Mechanics, Lund University, Lund, Sweden.

Balfour, J.A.D. (1986). *Computer Analysis of Structural Frameworks*. Collin, London.

Basu, S. (2011). *FreeMat v4.1 documentation*. http://freemat.sf.net/FreeMat-4.1.pdf , diakses 3 April 2013

Ferreira, A.J.M. (2009). *Matlab codes for finite element analysis*. Springer, Berlin.

Freemat (2013). www.freemat.org, diakses 3 April 2013

Kanok-Nukulchai, W. (1993). "*AIT1993".* Asian Institute of Technology, Bangkok.

Kassimali, A. (2012). Matrix analysis of structures. Cengage Learning, Satmford, USA.

Krishnamoorthy, C.S (1987). "*Finite Element Analysis"*, Tata McGraw-Hill Publishing Company Limited. New Delhi.

Mathworks Inc.(1996). "*Matlab the Language of Technical Programming"* Natick, MA.

Octave (2013). www.octave.org, diakses 3 April 2013

Scilab (2013). www.scilab.org, diakses 3 April 2013

Shim, J.S. (1980). "*Development of a Symbolic Manipulation Program*." *M.Eng. thesis.* Asian Institute of Technology. Bangkok.

Wilson, E.L. (1979). CAL-A Computer Analysis Languange for teaching structural analysis. *Computers and Structures,* Vol.10.

Wilson, E.L. (1986). "*CAL86: Computer Assisted Learning of Structural Analysis and the CAL/SAP Development System*." *Report no. UCB/SESM-86/05.* Department of Civil Engineering, University of California, Berkeley, California.

servient in lumine veritatis