19 5-d 22

DEPARTEMEN MATEMATIKA
FMIPA - INSTITUT PERTANIAN BOGOR

ISSN:
1412-677X

Journal of Mathematics and Its Applications

JMA

Jurnal Matematika dan Aplikasinya

Volume 12, No. 1
Juli 2013

Alamat Redaksi :
Departemen Matematika
FMIPA –Institut Pertanian Bogor
Jln. Meranti, Kampus IPB
Dramaga - Bogor

Phone/Fax: (0251) 8625276
E-mail: math@ipb.ac.id
jma.mathipb@gmail.com

Vol. 12, No. 1, Juli 2013 **ISSN : 1412-677X**

Journal of Mathematics and Its Applications

JMA

Jurnal Matematika dan Aplikasinya

ALAMAT REDAKSI:
Departemen Matematika
FMIPA – Institut Pertanian Bogor
Jln. Meranti, Kampus IPB Dramaga Bogor
Phone/Fax: (0251) 8625276
Email: math@ipb.ac.id, jma.mathipb@gmail.com
Website: www.math.ipb.ac.id/ojs

JMA merupakan media yang memuat informasi hasil penelitian matematika baik murni maupun terapan, bagi para matematikawan atau para pengguna matematika. JMA diterbitkan dua kali (dua nomor) setiap tahun (periode Juli dan Desember).

Harga langganan per volume, termasuk biaya pos:
Institusi/Perpustakaan Rp. 350.000,- (dalam IPB), Rp. 500.000,- (luar IPB)
Staf/Perorangan Rp. 200.000,- (dalam IPB), Rp.250.000,- (luar IPB)
Mahasiswa Rp. 75.000,-
Penulis makalah yang diterima dikenai biaya administrasi Rp.25.000,- per halaman

Semua pembayaran biaya dapat ditransfer melalui:

Nur Aliatiningtyas, Dra
BNI Cabang Bogor
No. Rek. 0254402360

𝔍𝔐𝔄, Vol. 12, No.1, Juli 2013 ISSN: 1412-677X

Journal of Mathematics and Its Applications

𝔍𝔐𝔄

Jurnal Matematika dan Aplikasinya

DAFTAR ISI

SIMULASI WAVEGUIDE SEDERHANA MENGGUNAKAN METODE GALERKIN DALAM MATLAB

ALATAS, H.[a,1], A.D. GARNADI[a,2], S. NURDIATI[2], T. PUJANEGARA[a], L. YULIAWATI[a]

Abstrak

Tulisan ini, merupakan sebuah tutorial bagaimana mengimplementasikan metode Galerkin untuk menyelesaikan persamaan Helmholtz. Persamaan ini, misalnya digunakan untuk memodelkan persamaan gelombang skalar. Misalnya untuk memperoleh informasi perilaku waveguide, persamaan Helmholtz perlu diselesaikan secara numerik, mengingat bentuk geometrik bahan penyusun menyebabkan solusi analitik sulit didapat. Persamaan gelombang skalar untuk waveguide isotropik homogen digunakan untuk memperkenalkan FEM. Untuk lebih sederhananya, digunakan elemen segitiga orde pertama. Makalah ini menunjukkan bagaimana pengetahuan tentang metode elemen hingga (FEM) dapat dipelajari dalam waktu singkat dengan menggunakan MATLAB. Hal ini menunjukkan bagaimana pengetahuan yang diperoleh dapat diperluas untuk masalah elektromagnetik lainnya.

PENDAHULUAN

Ketersediaan daya komputasi yang besar dan murah menggunakan komputer desktop atau laptop menjadikan solusi numerik dari permasalahan elektromagnetik menjadi hal yang layak, bahkan bagi mahasiswa sarjana sekalipun. Di kalangan pendidik telah diambil dua pendekatan: menggunakan perangkat lunak yang tersedia secara komersial [1] (yang mungkin menjadi pilihan yang mahal), atau desain antarmuka pengguna dan kode simulasi [2.3] berdasarkan paket matematis terprogram. Kedua pendekatan ini bukan merupakan obyek dari tulisan ini. Tujuan dari tulisan ini adalah memperkenalkan metode momen melalui Matlab dan menyelesaikan permasalahan elektromagnetik. Matlab telah digunakan di seluruh dunia dalam pengajaran banyak mata kuliah rekayasa, misalnya, pemrosesan sinyal dan teknik kontrol. Ini tidak akan mudah bagi para pengajar dalam bidang elektromagnetik untuk mengharapkan siswa untuk memiliki pengetahuan dan akses menggunakan Matlab. Metode Elemen Hingga (FEM) adalah teknik yang relatif mapan dalam elektromagnetik dan masih merupakan area penelitian yang cukup aktif. Hal ini didukung oleh beberapa buku teks dan monograf yang cukup baik tersedia. Tetapi, bahan tersebut hanya cocok bagi para peneliti atau untuk perkuliahan khusus. Selain itu, buku teks tersebut

[a] Research Cluster for Dynamics and Complex System, Fakultas Ilmu Pengetahuan Alam, Jalan Meranti Kampus IPB Dramaga Bogor, 16680.

[1] Departemen Fisika, Fakultas Ilmu Pengetahuan Alam, Jalan Meranti Kampus IPB Dramaga Bogor, 16680.

[2] Departemen Matematika, Fakultas Ilmu Pengetahuan Alam, Jalan Meranti Kampus IPB Dramaga Bogor, 16680.

$$\oint_C H \bullet dl = \int_S J \bullet n \, dS. \tag{2}$$

Pada tahun 1831 Faraday membuktikan fenomena perubahan medan magnet yang menimbulkan arus listrik. Hukum Faraday menyatakan bahwa perubahan medan listrik mengakibatkan tegangan di sepanjang loop tertutup C. Pada saat induksi magnet B yang menembus loop tertutup C berubah maka besaran tersebut sama dengan berkurangnya tegangan sesuai dengan perubahan waktu di loop tertutup C tersebut. Fenomena ini dapat dinyatakan oleh

$$\oint_C E \bullet dl = -\frac{\partial}{\partial t}\int_S B \bullet n \, dS. \tag{3}$$

Hukum integral keliling Ampere dan hukum Faraday dapat digabungkan dengan menggunakan perubahan listrik. Hal ini dapat diilustrasikan oleh arus listrik AC yang dialirkan ke kondensator. Di dalam kondensator tersimpan muatan listrik yang mengalami perubahan menurut waktu sehingga kondensator mengalirkan arus listrik. Rasio perubahan waktu terhadap perubahan listrik D disebut sebagai perubahan arus listrik yang sama dengan arus listrik dalam hukum Ampere. Akibatnya persamaan (2) menjadi

$$\oint_C H \bullet dl = \int_S J \bullet n \, dS + \frac{\partial}{\partial t}\int_S D \bullet n \, dS. \tag{4}$$

Hukum integral keliling Ampere ini dapat digabungkan dengan hukum Faraday dan persamaan tersebut merupakan persamaan dasar Maxwell. Dengan menggunakan teori Stokes, ruas kiri pada persamaan (3) dan (4) dapat diubah menjadi integral permukaan. Permukaan yang diintegralkan harus diambil sekecil mungkin supaya berlaku di seluruh ruang. Akibatnya berlaku penurunan persamaan Maxwell berikut

$$\nabla \times H = J + \frac{\partial D}{\partial t} \tag{5}$$

$$\nabla \times E = -\frac{\partial B}{\partial t}. \tag{6}$$

Dalam hukum dasar elektromagnet nilai muatan listrik sama dengan jumlah perubahan listrik yang ditimbulkannya dan dikenal sebagai hukum Gauss untuk perubahan listrik. Hukum Gauss yang lain yang berhubungan dengan perubahan listrik yaitu tidak ada fenomena muatan listrik yang hanya mempunyai satu kutub saja. Kedua fenomena di atas dapat digambarkan oleh persamaan di bawah ini

$$\nabla \bullet D = \rho \tag{7}$$

$$\nabla \bullet B = 0 \tag{8}$$

Persamaan (5) hingga 8) memuat banyak variabel diantaranya variabel medan listrik E, perubahan listrik D, medan listrik H dan induksi magnet B. Untuk mendapatkan medan listrik dan medan magnet akan dilakukan penurunan pada persamaan-persamaan tersebut.

Suatu medium berdielektrik yang homogen digambarkan di seluruh ruang analisa dengan permitivitas ε dan permeabilitas μ. Pada medium homogen, dengan $k < 0, k = -l^2$ dan u sebagai variabel medan listrik atau medan magnet.

FEM memecahkan persamaan (17) dengan meminimumkan fungsional bentuk lemah yang diberikan oleh:

$$F(u,u) = \int_{\Omega} (\nabla u \cdot \nabla u + kuu)\, dx. \tag{18}$$

Gelombang datar adalah gelombang ideal dimana gelombang elektromagnet ini terhantar di ruang bebas dengan kecepatan cahaya. Oleh karena itu, gelombang datar merupakan gelombang di lokasi tak terhingga dari sumber gelombang yang menimbulkannya. Pada saat gelombang datar terhantar di ruang yang homogen, amplitudo gelombang tidak akan berubah dan fasenya tidak mengalami ketidakkontinyuan. Tetapi pada saat gelombang datar melewati medium yang mempunyai tetapan medium yang berlainan maka komponen medan listrik dan medan magnet di perbatasan antar medium tersebut harus memenuhi syarat batas. Untuk kemudahan, kita perhatikan beberapa kasus berikut.

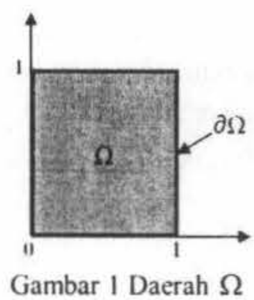
Kasus 1

Perhatikan persamaan Helmholtz dengan syarat batas Dirichlet berikut

$$\begin{aligned} -\nabla^2 u + ku &= 0 \text{ di } \Omega \\ u &= g \text{ di } \partial\Omega \end{aligned} \tag{19}$$

di mana k dan g adalah sembarang fungsi yang diketahui. Hal ini dapat diilustrasikan untuk gelombang datar yang melewati medium kedua yang merupakan penghantar sempurna.

Misalkan $\Omega = [0,1] \times [0,1]$ dan syarat batas $\partial\Omega$ yang diilustrasikan pada Gambar 1.



Gambar 1 Daerah Ω

Kasus 2

Hal lain yang mungkin jika gelombang datar tersebut mengalami perubahan setelah melewati medium kedua. Perhatikan persamaan Helmholtz dengan syarat batas Dirichlet dan Neumann berikut

$$A_{i,j} = \int_{\Omega} \nabla\varphi_j \nabla\varphi_i \, dx + \int_{\Omega} k\varphi_j \varphi_i \, dx \tag{24}$$

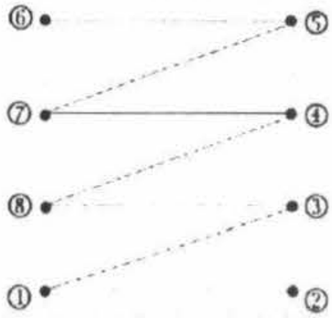
$$b_i = 0 \tag{25}$$

$$z = (z_1, z_2, ..., z_N)^T . \tag{26}$$

Pada persamaan di atas j bergerak sepanjang node dan i bergerak sepanjang node dengan nilai u diketahui. Nilai u pada $\partial\Omega$ yang diketahui dinotasikan dalam vektor z_e dan nilai u selainnya dinotasikan dalam z_n,

$$\left(A_e \mid A_n\right)\begin{pmatrix} z_e \\ z_n \end{pmatrix} = A_e z_e + A_n z_n = b$$

matriks A_e dihitung pada syarat batas. Sehingga solusi akhir z_n dapat dihitung dengan sistem persamaan $A_n z_n = b - A_e z_e$ dimana A_n sudah diketahui. Misalkan daerah dibagi ke dalam enam ($N = 6$) segitiga yang diperlihatkan pada Gambar 2.



Gambar 2 Ilustrasi mesh (enam elemen segitiga) dalam Ω

Misalkan koordinat dari setiap node diberikan pada tabel 1.

TABEL 1
Nomor-nomor koordinat sebagai node

Node	1	2	3	4	5	6	7	8
x	0	1	1	1	1	0	0	0
y	0	0	1/3	2/3	1	1	2/3	2/3

Matlab mampu membaca data dari file yang diberikan dalam format ascii dengan file .dat, sehingga dalam Matlab kita dapat menyimpan data koordinat untuk setiap node dengan pemrograman sebagai berikut.

```
nodel = [1 0 0;
     2 1 0;
     3 1 1/3;
     4 1 2/3;
     5 1 1;
     6 0 1;
     7 0 2/3;
     8 0 1/3];
```

Menyusun matriks *stiffness*

Perhatikan bahwa untuk sebuah elemen segitiga T misalkan (x_1, y_1), (x_2, y_2), dan (x_3, y_3) adalah titik-titik verteks dan φ_1, φ_2, dan φ_3 adalah fungsi basis yang berkorespondensi di K, yaitu

$$\varphi_i\left(x_j, y_j\right) = \delta_{ij}, \qquad i, j = 1,2,3$$

Oleh karena itu, diperoleh

$$\varphi_i(x,y) = \frac{\det\begin{pmatrix} 1 & x & y \\ 1 & x_{i+1} & y_{i+1} \\ 1 & x_{i+2} & y_{i+2} \end{pmatrix}}{\det\begin{pmatrix} 1 & x_i & y_i \\ 1 & x_{i+1} & y_{i+1} \\ 1 & x_{i+2} & y_{i+2} \end{pmatrix}}$$

dengan

$$\nabla\varphi_i(x,y) = \frac{1}{2|T|}\begin{pmatrix} y_{i+1} - y_{i+2} \\ x_{i+2} - x_{i+1} \end{pmatrix}. \tag{28}$$

Indeks yang digunakan pada persamaan di atas di dalam modulo 3, sehingga diperoleh

$$2|T| = \det\begin{pmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{pmatrix}.$$

Matriks *stiffness* yang diperoleh adalah

$$\begin{aligned} M_{ij} &= \int_T \nabla\varphi_i\left(\nabla\varphi_j\right)^T dx \\ &= \frac{|T|}{(2|T|)^2}\begin{pmatrix} y_{i+1} - y_{1+2} & x_{i+2} - x_{i+1} \end{pmatrix}\begin{pmatrix} y_{j+1} - y_{j+2} \\ x_{j+2} - x_{j+1} \end{pmatrix}. \end{aligned}$$

dengan indeks di dalam modulo 3. Bentuk sederhana persamaan di atas adalah

$$M = \frac{|T|}{2}GG^T,$$

di mana

$$G = \begin{pmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{pmatrix}^{-1}\begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

Pemrograman dalam Matlab untuk menyusun matriks *stiffness* dapat kita tuliskan sebagai berikut.

```
function M = stima(vertices)
d = size(vertices,2);
G = [ones(1,d+1);vertices'] \ [zeros(1,d);eye(d)];
```

```
load koordinat1.dat; koordinat1(:,1)=[]; % koordinat-
koordinat
load elemen1.dat; elemen1(:,1)=[]; % elemen hingga berbentuk
segitiga
load deltaomega.dat; deltaomega(:,1)=[];
koordinat = koordinat1;
elemen = elemen1;
for k = 1:3
[Kantennr,elemen] = GeneriereKantennr(elemen,koordinat);
VK = (1:full(max(max(Kantennr))))' + size(koordinat,1);
[koordinat,elemen] = ...
Verfeinerung(koordinat,elemen,[],[],Kantennr,VK);
end
FreeNodes=setdiff(1:size(koordinat,1),unique(deltaomega));
A = sparse(size(koordinat,1),size(koordinat,1));
b = sparse(size(koordinat,1),1);
% 1. menyusun matriks A1
% dengan memanggil fungsi stima
for i = 1:size(elemen,1)
A(elemen(i,:),elemen(i,:)) = A(elemen(i,:),elemen(i,:)) ...
+ stima(koordinat(elemen(i,:),:));
end
% 2. menyusun matriks A2
for i = 1:size(elemen,1)
A(elemen(i,:),elemen(i,:)) = A(elemen(i,:),elemen(i,:))+...
det([ones(1,3);koordinat(elemen(i,:),:)'])*...
(diag(ones(3,1))+
ones(3))*k0(sum(koordinat(elemen(i,:),:))/3)/24;
end
u = sparse(size(koordinat,1),1);
u(unique(deltaomega)) =
deltaO(koordinat(unique(deltaomega),:));
b = b - A*u;
% 3. Computation of the solution
u(FreeNodes) = A(FreeNodes,FreeNodes) \ b(FreeNodes);
% 4. Graphic representation
show(elemen,[],koordinat,full(u));
xlabel('x');ylabel('y');zlabel('u');
```

Kasus 2

Bentuk lemah untuk kasus 2 adalah

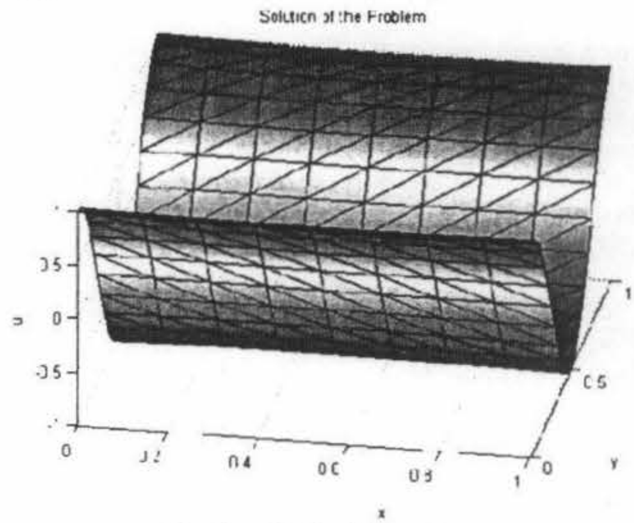
$$\int_{\Omega}\left(\nabla u\cdot\nabla v+kuv\right)dx=\int_{L_2} qv\,ds. \qquad (29)$$

Pendekatan elemen hingga (*finite element approximation*) dari bentuk lemah persamaan (29) adalah $u_h \in V_h$ sedemikian sehingga

Misalkan dipilih $k = -(2\pi)^2$, $p = \cos(2\pi y)$ dan $q = 0$.

```
function k = k1(x,y);
k = -((2*pi))^2;
end
function S = g(x)
S = zeros(size(x,1),1);
end
function a=q1(x);
a=0;
end
function BatasDirichletKasus2a = ud1(x)
BatasDirichletKasus2a = zeros(size(x,1),1);
I=find(x(:,1)==0|x(:,1)==1);
BatasDirichletKasus2a (I) = cos(2*pi*x(:,2));
end
```

Dengan menggunakan Matlab diperoleh solusi pada Gambar 5.



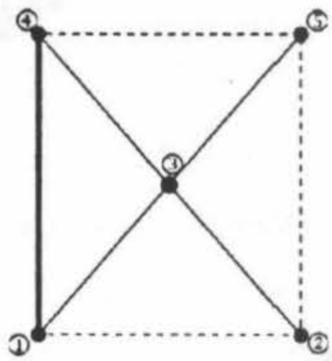
Gambar 5 Solusi kasus 2a

Pemrograman dengan Matlab sebagai berikut:

```
% inisialisasi
load koordinat1.dat; koordinat1(:,1)=[]; % koordinat-
koordinat
load elemen1.dat; elemen1(:,1)=[]; % elemen hingga
berbentuk segitiga
load neumann1.dat; neumann1 (:,1)=[];
load dirichlet1.dat; dirichlet1 (:,1)=[];
koordinat = koordinat1;
elemen = elemen1;
```

```
% 7. Graphic representation
show(elemen,[],koordinat,full(u));
xlabel('x');ylabel('y');zlabel('u');
```

b. Misalkan daerah dibagi ke dalam empat (N = 4) segitiga yang diperlihatkan pada gambar berikut ini Dengan nomor-nomor node, elemen segitiga dan edge syarat batas pada tabel berikut: Misalkan $k = -(2\pi)^2, p = \cos(2\pi y)$ dan q = 0, dengan menggunakan Matlab, diperoleh solusi dalam bentuk grafik berikut ini:



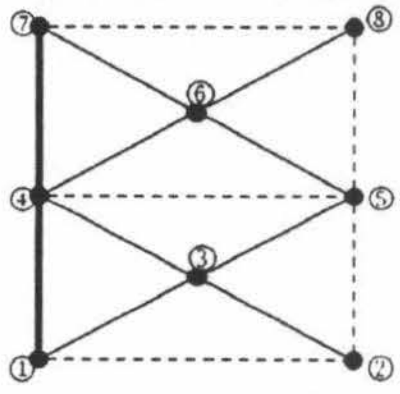
Gambar 6 Daerah dibagi ke dalam empat segitiga

TABEL 4
Nomor-nomor node, elemen, dan edge syarat batas

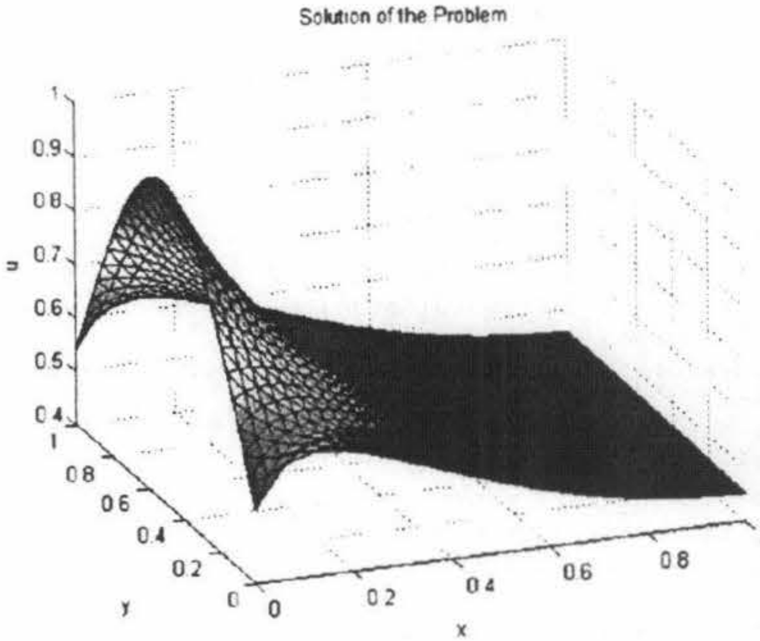
Nomor	Node (koordinat)	Elemen Segitiga	Syarat Batas Neumann	Syarat Batas Dirichlet
1	(0,0)	1-2-3	1-2	4-1
2	(1,0)	1-3-4	2-5	
3	(1/2,1/2)	3-2-5	5-4	
4	(0,1)	3-5-4		
5	(1,1)			

```
A(elemen(i,:),elemen(i,:)) =
A(elemen(i,:),elemen(i,:))+...
det([ones(1,3);koordinat(elemen(i,:),:)'])*...
(diag(ones(3,1))+
ones(3))*k1(sum(koordinat(elemen(i,:),:))/3)/24;
end
% 3. volume forces
for i = 1:size(elemen,1)
b(elemen(i,:)) = b(elemen(i,:))+...
det([ones(1,3);koordinat(elemen(i,:),:)'])*...
g(sum(koordinat(elemen(i,:))/3))/6;
end
% 4. Neumann condition
for i = 1:size(neumann,1)
b(neumann(i,:))=b(neumann(i,:)) +...
norm(koordinat(neumann(i,1),:)- ...
koordinat(neumann(i,2),:)) *...
q1(sum(koordinat(neumann(i,:),:))/2)/2;
end
% 5. Dirichlet conditions
u = sparse(size(koordinat,1),1);
u(unique(dirichlet)) =
ud1(koordinat(unique(dirichlet),:));
b = b - A*u;
% 6. Computation of the solution
u(FreeNodes) = A(FreeNodes,FreeNodes) \ b(FreeNodes);
% 7. Graphic representation
show(elemen,[],koordinat,full(u));
xlabel('x');ylabel('y');zlabel('u');
```

c. Misalkan daerah dibagi ke dalam delapan ($N = 8$) segitiga yang diperlihatkan pada gambar berikut ini. Dengan nomor-nomor node, elemen



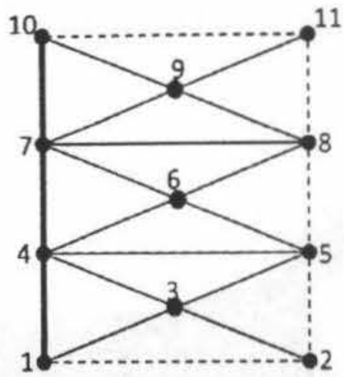
Gambar 8 Daerah dibagi ke dalam delapan segitiga



Gambar 9 Solusi Kasus 2c

Pemrograman dengan Matlab sebagai berikut:

```
load koordinat3.dat; koordinat3(:,1)=[]; % koordinat-
koordinat
load elemen3.dat; elemen3(:,1)=[]; % elemen hingga
berbentuk segitiga
load neumann4.dat; neumann4(:,1)=[];
load dirichlet4.dat; dirichlet4(:,1)=[];
koordinat=koordinat3;
elemen=elemen3;
neumann = neumann4;
dirichlet = dirichlet4;
for k = 1:4
[Kantennr,elemen] = GeneriereKantennr(elemen,koordinat);
VK = (1:full(max(max(Kantennr))))' + size(koordinat,1);
[koordinat,elemen,dirichlet,neumann] = ...
Verfeinerung(koordinat,elemen,dirichlet,neumann,Kantennr
,VK);
end
20
FreeNodes=setdiff(1:size(koordinat,1),unique(dirichlet))
;
A = sparse(size(koordinat,1),size(koordinat,1));
b = sparse(size(koordinat,1),1);
% 1. menyusun matriks A1
% dengan memanggil fungsi stima_1
```



Gambar 10 Daerah Ω dibagi ke dalam duabelas segitiga

TABEL 6
Nomor-nomor node, elemen, dan edge syarat batas

Nomor	Node (koordinat)	Elemen Segitiga	Syarat Batas Neumann	Syarat Batas Dirichlet
1	(0,0)	1-2-3	1-2	10-7
2	(1,0)	1-3-4	2-5	7-4
3	(1/2,1/6)	3-2-5	5-8	4-1
4	(0,1/3)	3-5-4	8-11	
5	(1,1/3)	4-5-6	11-10	
6	(1/2,1/2)	4-6-7		
7	(0,2/3)	6-5-8		
8	(1,2/3)	6-8-7		
9	(1/2,5/6)	7-8-9		
10	(0,1)	7-9-10		
11	(1,1)	9-8-11		
12		9-10-11		

```
function k = k4(x)
if (x(:,2)<1/3)||(x(:,2)>2/3)
k = 1.4;
else
k = 1.5;
end
end
```

Dengan menggunakan Matlab, diperoleh solusi dalam bentuk grafik berikut ini:

```
A(elemen(i,:),elemen(i,:)) = A(elemen(i,:),elemen(i,:)) ...
+ stima(koordinat(elemen(i,:),:));
end
% 2. menyusun matriks A2
for i = 1:size(elemen,1)
A(elemen(i,:),elemen(i,:)) = A(elemen(i,:),elemen(i,:))+...
det([ones(1,3);koordinat(elemen(i,:),:)'])*...
(diag(ones(3,1))+
ones(3))*k3(sum(koordinat(elemen(i,:),:))/3)/24;
end
% 3. volume forces
for i = 1:size(elemen,1)
b(elemen(i,:)) = b(elemen(i,:))+...
det([ones(1,3);koordinat(elemen(i,:),:)'])*...
g(sum(koordinat(elemen(i,:))/3))/6;
end
% 4. Neumann condition
for i = 1:size(neumann,1)
b(neumann(i,:))=b(neumann(i,:)) +...
norm(koordinat(neumann(i,1),:)- ...
koordinat(neumann(i,2),:)) *...
q1(sum(koordinat(neumann(i,:),:))/2)/2;
end
% 5. Dirichlet conditions
u = sparse(size(koordinat,1),1);
u(unique(dirichlet)) = ud3(koordinat(unique(dirichlet),:));
b = b - A*u;
% 6. Computation of the solution
u(FreeNodes) = A(FreeNodes,FreeNodes) \ b(FreeNodes);
% 7. Graphic representation
show(elemen,[],koordinat,full(u));
xlabel('x');ylabel('y');zlabel('u');
```

SIMPULAN

Kode Matlab berhubungan langsung dengan operasi FEM. Ditunjukkan bagaimana FEM diperkenalkan kepada mahasiswa hanya menggunakan beberapa baris kode Matlab. Kedua contoh yang diberikan dapat dijalankan pada PC standar. Hal ini juga menunjukkan bagaimana konsep-konsep yang diperoleh dapat dengan mudah diperluas ke masalah lain yang diberikan bersifat 2D. Namun, teknik pemrograman yang sama dapat digunakan untuk masalah yang lebih kompleks, seperti masalah 3D yang misalnya disajikan oleh Silvester dan Ferrari [6]. Lembaga pendidikan dengan sumber daya keuangan yang terbatas dapat mengambil manfaat dari metodologi yang diberikan, karena hanya memerlukan Matlab, tidak ada add-ons (bahkan pembangkit mesh sekalipun). Materi yang