

BAB VI

KESIMPULAN DAN SARAN

6.1. Kesimpulan

Setelah melalui tahap analisis dan komparasi hasil analisis, maka diperoleh beberapa kesimpulan sebagai berikut:

1. Ada kekurangan dalam analisis eksak yaitu adanya asumsi rasio redaman 2% dianggap redaman Rayleigh yaitu redaman sebanding massa dan kekakuan. Hal ini menyebabkan kesulitan dalam komparasi rasio redaman karena salah satunya berupa asumsi dan yang lain adalah hasil analisis. Perlu dipelajari lebih lanjut cara menghasilkan rasio redaman dari sifat-sifat struktur itu sendiri.
2. Identifikasi parameter dinamik dengan metode DD-SSI menghasilkan frekuensi alami yang sangat akurat sesuai dengan hasil analisis eksak baik untuk konfigurasi struktur utuh maupun struktur rusak masing-masing dengan tiga model struktur yang berbeda dari input respon percepatan struktur. Namun untuk kasus rangka batang dengan 12 derajat kebebasan metode DD-SSI hanya dapat mengidentifikasi frekuensi alami dan rasio redaman di empat mode pertama saja. Hal ini menimbulkan gagasan untuk mereduksi derajat kebebasan struktur rangka batang dengan metode kondensasi statik, namun dari sumber Paz dan Leigh (2004) dalam analisis dinamik rangka batang tidak dilakukan kondensasi statik.

3. Identifikasi parameter dinamik dengan metode DD-SSI melibatkan penggunaan diagram stabilisasi untuk memplotkan hasil perhitungan dengan metode DD-SSI baik frekuensi alami maupun rasio redaman sehingga dapat dibedakan hasil perhitungan yang stabil maupun tidak stabil. Untuk diagram stabilisasi frekuensi alami pola-pola stabil dan tidak stabil lebih mudah dibedakan dan lebih mudah dipilih, sedangkan untuk diagram stabilisasi rasio redaman pola-pola itu kadang tidak terbentuk dan sulit untuk membedakan maupun memilih pola stabil mana untuk dijadikan hasil dari analisis.
4. Metode DD-SSI cukup mudah untuk dimengerti dan diaplikasikan, namun butuh waktu dan kesabaran yang lebih banyak untuk menghasilkan parameter dinamik yang stabil untuk dipilih sebagai hasil analisis. Hal ini disebabkan metode DD-SSI harus melalui tahap *trial* dengan input ordo maksimum sistem (n) dan besaran blok baris (x). Kedua input ini harus ditrial sampai hasil pada diagram stabilisasi membentuk pola-pola stabil.
5. Hasil perbandingan parameter dinamik yang dihasilkan DD-SSI pada struktur yang utuh dan rusak menunjukkan terjadinya penurunan frekuensi alami struktur pada struktur yang rusak atau diperlemah. Hal ini sesuai dengan penelitian Peeters dkk (1996;1999a) yang menerapkan metode DD-SSI pada eksperimen balok beton bertulang yang telah terjadi *crack* menghasilkan penurunan frekuensi alami 7-18% pada struktur yang rusak. Rasio perbedaan yang terlalu besar antara konfigurasi utuh dan rusak menyimpulkan bahwa asumsi skenario kerusakan dengan mengurangi

dimensi tampang elemen struktur sebesar 50% dari struktur utuhnya, terlalu besar. Pengurangan dimensi tampang elemen untuk mereduksi kekuatan struktur sangat efektif mereduksi kekuatan struktur.

5.2. Saran

Berdasarkan proses, hasil dan kesimpulan dari penelitian ini maka penulis menyarankan beberapa hal yang sekiranya penting, yaitu:

1. Diperlukan metode pembandingan untuk membuktikan keakuratan hasil metode DD-SSI baik dalam domain frekuensi seperti *Peak Picking* (PP), *Frequency Domain Decomposition* (FDD) juga dalam domain waktu seperti metode SSI yang lebih tua yaitu *Covariance Stochastic Subspace Identification* (Cov-SSI).
2. Perlu dilakukan penelitian lanjutan dengan melakukan eksperimen pada model simulasi struktur atau pada struktur sebenarnya untuk mengetahui kemampuan dan aplikasi metode DD-SSI dalam pengujian eksperimen.

DAFTAR PUSTAKA

- Arfiadi, Y., 2016, *Analisis Struktur dengan Program Matlab dan Freemat*, Cahaya Atma Pustaka, Kelompok Penerbit Universitas Atma Jaya Yogyakarta, Yogyakarta.
- Beskhyroun, S., Ma, Q., 2012, *Low-Cost Accelerometers for Experimental Modal Analysis*, Proceedings of 15th World Conference on Earthquake Engineering Lisbon, Portugal.
- Brincker, R., 2014, *Some Elements of Operational Modal Analysis*, Journal of Shock and Vibration Vol. 2014, Article ID 325839.
- Chopra, A.K., 2011, *Dynamics of Structures - Theory and Applications to Earthquake Engineering*, Pearson Prentice Hall, 4th edition.
- Ghalishooyan, M., Shooshtari, A., 2015, *Operational Modal Analysis Techniques and Their Theoretical and Practical Aspect: A Comprehensive Review and Introduction*, Proceedings of 6th International Operational Modal Analysis Conference Gijon, Spain.
- Golub, G., Van Loan, C., 1996, *Matrix Computations*, The Johns Hopkins University Press, 3rd edition.
- Liao, S., Zerva, A., 2004, *Application of Subspace-Based Blind Identification Method in Structural System*, Proceedings of 13th World Conference on Earthquake Engineering Vancouver, B.C., Canada.
- Newland, D. E., 1996, *An Introduction to Random Vibrations, Spectral & Wavelet Analysis*, Prentice Hall, 3rd edition.
- Paz, M., 1986, *Microcomputer-Aid Engineering : Structural Dynamics*, Van Nostrand Reinhold, New York.
- Paz, M., Leigh, W., 2004, *Structural Dynamics : Theory and Computation 5th Edition Update with SAP 2000*, Kluwer Academic Publisher, Massachusetts, USA.
- Peeters, B., De Roeck, G., Pollet, T., Schueremans, L., 1995, *Stochastic Subspace Techniques Applied to Parameter Identification of Civil Engineering Structures*, Proceedings of New Advances in Modal Synthesis of Large Structures: Nonlinear, Damped and Nondeterministic Cases, pp. 151-162, Lyon, France.

- Peeters, B., Wahab, A.M., De Roeck, G., De Visser, J., De Wilde, P.W., Ndambi, M., Vantomme, J., 1996, *Evaluation of Structural Damage by dynamic system identification*, Proceedings of 21th International Seminar on Modal Analysis, pp. 1349-1361, Katholieke Universiteit Leuven, Belgium.
- Peeters, B., De Roeck, G., 1997, *The Performance of Time Domain System Identification Methods Applied to Operational Data*, Proceedings of Structural Damage Assessment Using Advanced Signal Processing Procedures, pp. 377-386, University of Sheffield, Sheffield, UK.
- Peeters, B., De Roeck, G., L. Hermans, T. Wauters, C. Kramer, C.A.M. de Smet, 1998a, *Comparison of System Identification Methods Using Operational Data of A Bridge*, Proceedings of 23th International Seminar on Modal Analysis, pp. 923-930, Katholieke Universiteit Leuven, Belgium.
- Peeters, B., De Roeck, G., 1998b, *Stochastic Subspace System Identification of A Steel Transmitter Mast*, Journal of Department of Civil Engineering Katholieke Universiteit Leuven W. de Croylaan 2 B-3001 Heverlee, Belgium.
- Peeters, B., De Roeck, G., 1999(a), *Stochastic System Identification: Uncertainty of the Estimated Modal Parameters*, Proceedings of The International Modal Analysis Conference, pp. 231-237, Kissimmee, Florida, USA.
- Peeters, B., De Roeck, G., 1999(b), *Experimental Dynamic Analysis of A Steel Mast Excited by Wind Load*, Proceedings of the 4th European Conference on Structural Dynamic, pp. 1075-1080, Prague, Czech Republic.
- Peeters, B., De Roeck, G., 1999(c), *Referenced-Based Stochastic Subspace Identification for Output-Only Modal Analysis*, Journal of Mechanical System and Signal Processing Vol. 13 pp. 855-878.
- Peeters, B., 2000, *System Identification and Damage Detection in Civil Engineering*, PhD Thesis, Department of Civil Engineering, Katholieke Universiteit Leuven, Leuven, Belgium.
- Peeters, B., De Roeck, G., 2001, *Stochastic System Identification for Operational Modal Analysis: A Review*, Journal of Dynamics System, Measurement and Control, Vol. 123 pp. 659.
- Rainieri, C., 2008, *Operational Modal Analysis for Seismic Protection of Structures*, PhD Thesis, Department of Civil Engineering, University of Naples "Federico II", Naples, Italy.
- Rainieri, C., Fabbrocino, G., 2011, *Operational Modal Analysis for the Characterization of Heritage Structures*, Journal of Goefizika Vol. 28.

Rainieri, C., Fabbrocino, G., 2014, *Operation Modal Analysis of Civil Engineering Structures*, Springer, 1st edition.

Schanke, A.S., 2015, *Operational Modal Analysis of Large Bridges*, Master Thesis, Departement of Civil and Enviromental Engineering Norwegian University of science and Technology.

Van Overshce, P., De Moor, B., 1996, *Subspace Identification for Linear Systems: Theory – Implementation - Applications*. Kluwer Academic Publishers, 1st edition.



1. mlt.m

Editor - D:\ds\ds\mlt.m

```

1  function m=mlt(rho,L)
2  % mlt penyusunan matriks massa batang untuk struktur truss
3  % dalam koordinat lokal
4
5  %m=mlt(rho,L)
6  %rho = massa per meter panjang
7  %L  = panjang batang
8
9  % Happy Kurniawan 8-8-2017
10
11  ml=rho*L/6;
12  m=[2*ml 0 ml 0
13     0 2*ml 0 ml
14     ml 0 2*ml 0
15     0 ml 0 2*ml];

```

2. mg.m

Editor - D:\ds\ds\mg.m

```

1  function M=mg(m,T)
2  % mg penyusunan matriks massa batang
3  % dalam koordinat global
4
5  %M=mg(m,T)
6  %m = matriks massa koordinat lokal
7  %T = matriks transformasi
8
9  %Happy Kurniawan 8-8-2017
10
11  M=T'*m*T

```

3. memb.m

Editor - D:\ds\ds\memb.m

```

1 function [L,T]=memb(ni,nj)
2 % memb menyusun panjang elemen dan matriks transformasi
3 % untuk struktur balok sederhana
4 % [L,T]= memb(ni,nj)
5 % L = panjang elemen
6 % T = matriks transformasi
7 % ni = nomor titik awal elemen
8 % nj = nomor titik akhir elemen
9 % catatan : koordinat ni, nj sudah ditentukan di coor.m
10 % lihat juga memf.m dan memt.m
11
12 % dikembangkan oleh Happy Kurniawan
13 % Universitas Atma Jaya Yogyakarta
14
15 L=sqrt(sum((ni-nj).^2));
16 % L = sqrt((ci(1,1)-ni(1,1))^2 + (nj(1,2)-ni(1,2))^2));
17 c=(nj(1,1)-ni(1,1))/L;
18 s=(nj(1,2)-ni(1,2))/L;
19 T=[c 0 0 0;0 1 0 0;0 0 c 0;0 0 0 1]

```

4. klb.m

Editor - D:\ds\ds\klb.m

```

1 function k=klb(E,I,L)
2 % klb menyusun matriks kekakuan lokal batang untuk struktur balok
3 % k=klb(E,I,L)
4 % k = kekakuan dalam sumbu lokal
5 % E = modulus elastisitas
6 % I = momen inersia tampang batang
7 % L = panjang batang
8
9 % dikembangkan oleh Happy Kurniawan
10 % Universitas Atma Jaya Yogyakarta
11
12 k1=(2*E*I)/L^3;
13 k=[6*k1 3*L*k1 -6*k1 3*L*k1
14 3*L*k1 2*L^2*k1 -3*L*k1 L^2*k1
15 -6*k1 -3*L*k1 6*k1 -3*L*k1
16 3*L*k1 L^2*k1 -3*L*k1 2*L^2*k1];

```


5. mlb.m

Editor - D:\dsl\dsl\mlb.m

```

EDITOR      PUBLISH      VIEW
+ New      + Open      + Save      + Find Files      + Go To      + Insert      + Breakpoints      + Run
  |         |           |           | Compare      | Go To      | Comment      | Breakpoints      | Run
  |         |           |           | Print         | Find       | Indent       | Breakpoints      | Run
  |         |           |           |               |           |             | Breakpoints      | Run
FILE        NAVIGATE      EDIT        BREAKPOINTS

balokpazsolve.m x framesolve.m x truss12solve.m x SSIData.m x diagstab.m x

1 function m=mlb(rho,A,L)
2 % mlb penyusunan matriks massa batang untuk struktur balok
3 % dalam koordinat lokal
4
5 %m=mlb(rho,A,L)
6 %rho = massa jenis
7 %L = panjang batang
8 %A = luas penampang balok
9
10 % Happy Kurniawan 8-8-2017
11
12 ml=(rho*A*L)/420
13 m=[156*ml    22*L*ml    54*ml    -13*L*ml
14     22*L*ml    4*L^2*ml    13*L*ml    -3*L^2*ml
15     54*ml    13*L*ml    156*ml    -22*L*ml
16     -13*L*ml    -3*L^2*ml    -22*L*ml    4*L^2*ml]

```

6. assb.m

Editor - D:\dsl\dsl\assb.m

```

EDITOR      PUBLISH      VIEW
+ New      + Open      + Save      + Find Files      + Go To      + Insert      + Breakpoints      + Run      + Run and Advance
  |         |           |           | Compare      | Go To      | Comment      | Breakpoints      | Run      | Run and Advance
  |         |           |           | Print         | Find       | Indent       | Breakpoints      | Run      | Run and Advance
  |         |           |           |               |           |             | Breakpoints      | Run      | Run and Advance
FILE        NAVIGATE      EDIT        BREAKPOINTS      RUN

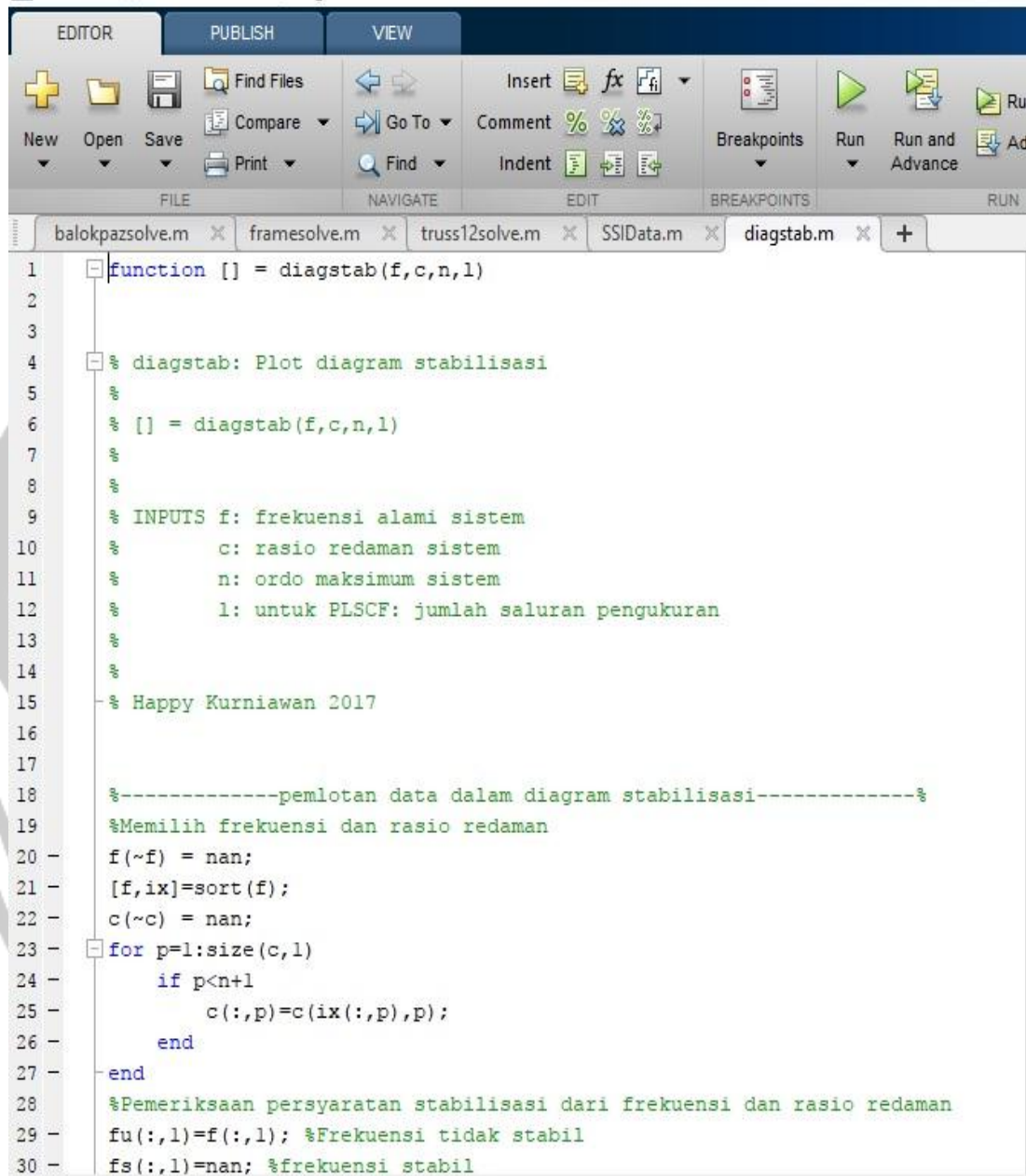
balokpazsolve.m x framesolve.m x truss12solve.m x SSIData.m x diagstab.m x assb.m x n

1 function ka=assb(kgf,id,dof)
2 % ka = assb(kgf,id,dof)
3 % menyusun matriks kekakuan struktur sesuai dengan vektor ID dan
4 % total derajat kebebasan struktur
5 % ka = kontribusi kgb terhadap matriks kekakuan struktur
6 % kgb = matriks kekakuan yang akan disusun
7 % ID = vektor ID
8 % dof = total derajat kebebasan struktur
9
10 % dikembangkan oleh Happy Kurniawan
11 % Universitas Atma Jaya Yogyakarta
12
13 ka=zeros(dof,dof);
14 idt=id';
15 for i=1:4
16     ii=id(1,i);
17     for j=1:4
18         ij=idt(j,1);
19         if(ii~=0 & ij~=0)
20             ka(ii,ij)=kgb(i,j);
21         end
22     end
23 end

```

1. diagstab.m

Editor - D:\SSI MATLAB CODE\diagstab.m



```

1 function [] = diagstab(f,c,n,l)
2
3
4 % diagstab: Plot diagram stabilisasi
5 %
6 % [] = diagstab(f,c,n,l)
7 %
8 %
9 % INPUTS f: frekuensi alami sistem
10 %      c: rasio redaman sistem
11 %      n: ordo maksimum sistem
12 %      l: untuk PLSCF: jumlah saluran pengukuran
13 %
14 %
15 % Happy Kurniawan 2017
16
17
18 %-----pemlotan data dalam diagram stabilisasi-----%
19 %Memilih frekuensi dan rasio redaman
20 f(~f) = nan;
21 [f,ix]=sort(f);
22 c(~c) = nan;
23 for p=1:size(c,l)
24     if p<n+1
25         c(:,p)=c(ix(:,p),p);
26     end
27 end
28 %Pemeriksaan persyaratan stabilisasi dari frekuensi dan rasio redaman
29 fu(:,l)=f(:,l); %Frekuensi tidak stabil
30 fs(:,l)=nan; %frekuensi stabil

```

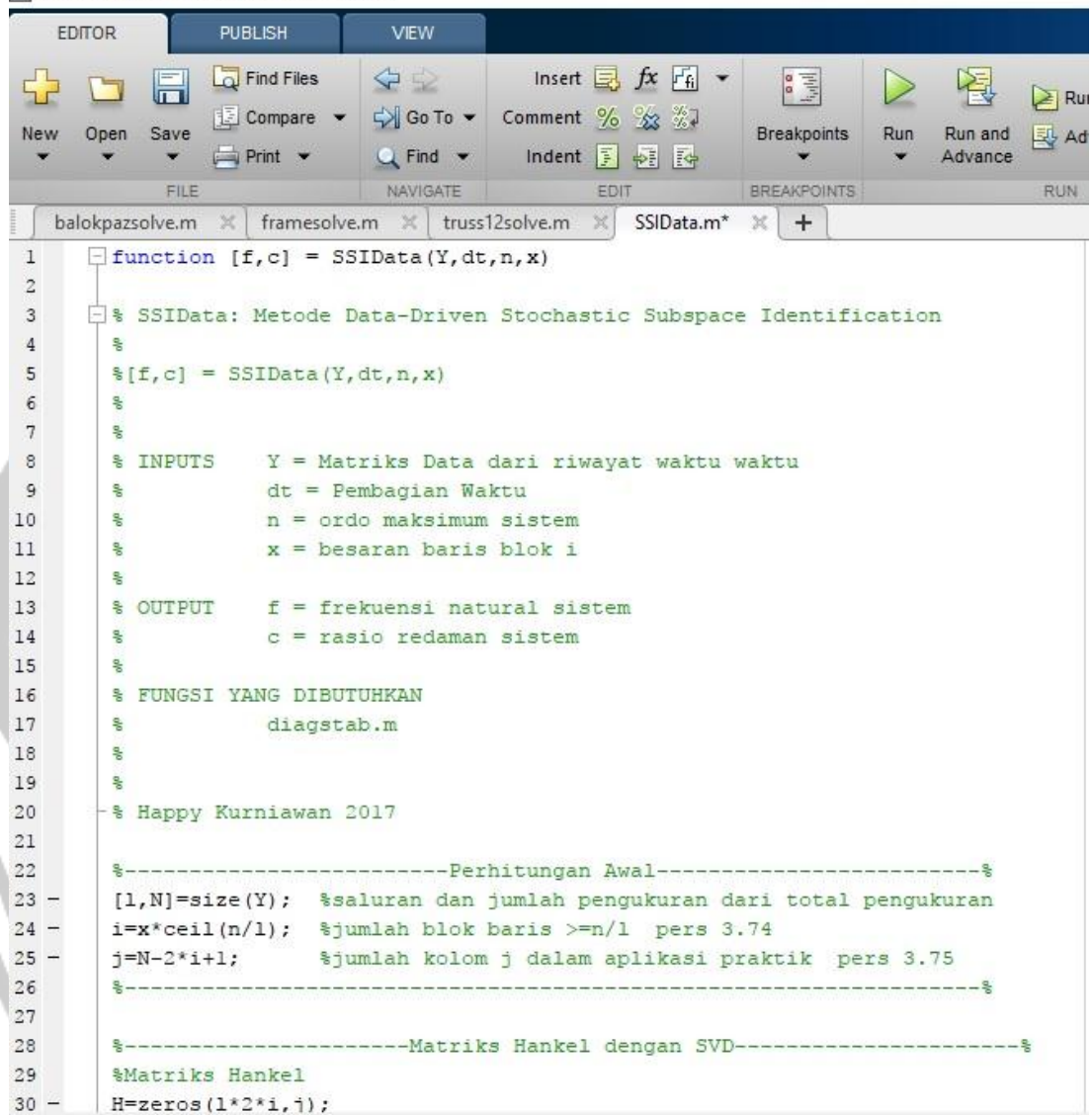
```

31 -   fcu(:,1)=f(:,1); %rasio redaman tidak stabil
32 -   fcs(:,1)=nan; %Rasio redaman stabil
33 -   for k=2:n
34 -       for j=1:n*1
35 -           ch=0;
36 -           chd=0;
37 -           for m=1:n*1
38 -               if ch == 0
39 -                   sf=abs(f(m,k-1)-f(j,k))/f(m,k-1); %pemeriksaan frekuensi
40 -                   if sf < 0.001 %Batas nilai frekuensi
41 -                       fs(j,k)=f(j,k);
42 -                       fu(j,k)=nan;
43 -                       ch=1;
44 -                   else
45 -                       fu(j,k)=f(j,k);
46 -                       fs(j,k)=nan;
47 -                       ch=0;
48 -                   end
49 -               end
50 -               if chd == 0
51 -                   sc=abs(c(m,k-1)-c(j,k))/c(m,k-1); %Pemeriksaan redaman
52 -                   if sc < 0.05 %nilai batas redaman
53 -                       fcs(j,k)=f(j,k);
54 -                       fcu(j,k)=nan;
55 -                       chd=1;
56 -                   else
57 -                       fcu(j,k)=f(j,k);
58 -                       fcs(j,k)=nan;
59 -                       chd=0;
60 -                   end
61 -               end
62 -           end
63 -       end
64 -   end
65 -   %Plotting
66 -   figure(1);
67 -   for k=1:n
68 -       scatter(fs(:,k),ones(1,n*1)*k,50,'blue');
69 -       scatter(fu(:,k),ones(1,n*1)*k,50,'red');
70 -       scatter(fcs(:,k),ones(1,n*1)*k,50,'blue','+');
71 -       scatter(fcu(:,k),ones(1,n*1)*k,50,'red','+');
72 -       hold on
73 -   end
74 -   hold off
75 -
76 -   xlabel('frekuensi (rad/s)');
77 -   ylabel('Ordo');
78 -   title('Diagram Stabilisasi');
79 -   %-----
80 -   end

```

2. SSIData.m

Editor - D:\SSI MATLAB CODE\SSIData.m*



```

1  function [f,c] = SSIData(Y,dt,n,x)
2
3  % SSIData: Metode Data-Driven Stochastic Subspace Identification
4  %
5  % [f,c] = SSIData(Y,dt,n,x)
6  %
7  %
8  % INPUTS   Y = Matriks Data dari riwayat waktu waktu
9  %          dt = Pembagian Waktu
10 %          n = ordo maksimum sistem
11 %          x = besaran baris blok i
12 %
13 % OUTPUT   f = frekuensi natural sistem
14 %          c = rasio redaman sistem
15 %
16 % FUNGSI YANG DIBUTUHKAN
17 %         diagstab.m
18 %
19 %
20 % Happy Kurniawan 2017
21
22 %-----Perhitungan Awal-----%
23 [l,N]=size(Y); %saluran dan jumlah pengukuran dari total pengukuran
24 i=x*ceil(n/l); %jumlah blok baris >=n/l pers 3.74
25 j=N-2*i+1;    %jumlah kolom j dalam aplikasi praktik pers 3.75
26 %-----%
27
28 %-----Matriks Hankel dengan SVD-----%
29 %Matriks Hankel
30 H=zeros(1*2*i,i);

```



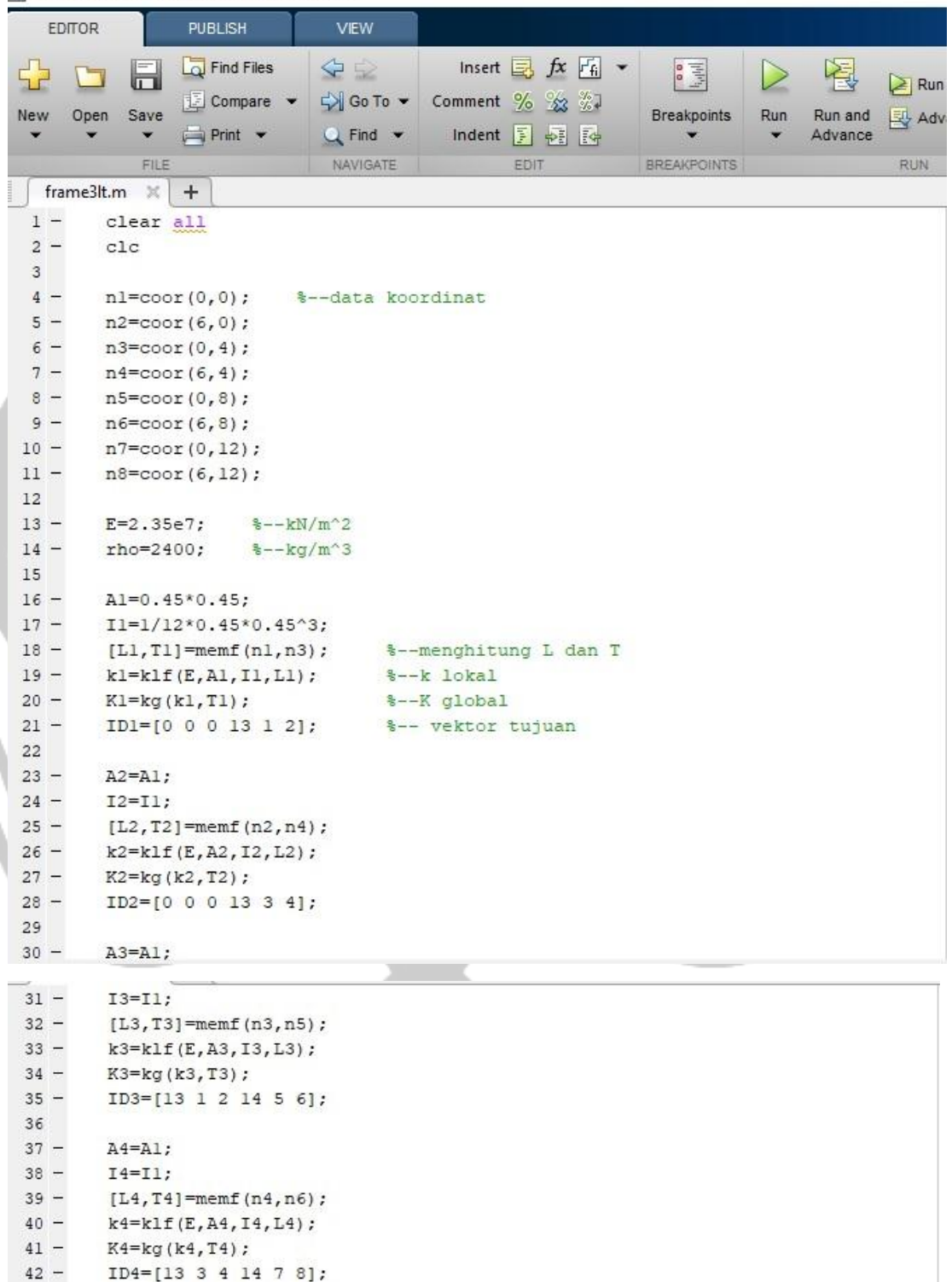
```

31 - for k=1:2*i
32 -     H((k-1)*l+1:k*l,:) = Y(:,k:k+j-1);    %pers 3.76
33 - end
34
35 %Faktorisasi LQ
36 [Q,L]=qr(H',0); %pers 3.77
37 L=L'; %Matriks Triangular yang lebih kecil L pers 3.78
38 Q=Q'; %Matriks Ortonormal Q pers 3.79
39
40 %Nilai dari Matriks L dan Q yang akan digunakan selanjutnya
41 L21=L(1*i+1:1*i+1,1:1*i);
42 L22=L(1*i+1:1*i+1,1*i+1:1*i+1);
43 L31=L(1*i+1+1:2*1*i,1:1*i);
44 L32=L(1*i+1+1:2*1*i,1*i+1:1*i+1);
45 Q1=Q(1:1*i,:);
46 Q2=Q(1*i+1:1*i+1,:);
47
48 %Proyeksi
49 Pi=[L21;L31]*Q1; %pers 3.80
50 Pim=[L31 L32]*[Q1;Q2]; %pers 3.81
51
52 %Output Sequence
53 Yi=[L21 L22]*[Q1;Q2]; %pers 3.82
54
55 %SVD
56 [U,S]=svd(Pi); %pers 3.83
57 s=diag(S);
58
59 clear L Q L21 L22 L31 L32 Q1 Q2 H S N Y
60 %-----%
62 %-----Perhitungan Matriks State A dan Parameter Modal untuk Setiap Ordo-----%
63 f=zeros(n,n); %Menentukan Efisiensi Perhitungan
64 c=zeros(n,n); %Menentukan Efisiensi Perhitungan
65 for k=1:n
66     Uk=U(:,1:k); %Menentukan untuk Perhitungan selanjutnya
67     ss=diag(sqrt(s(1:k))); %Menentukan untuk Perhitungan selanjutnya
68     Oi=Uk*ss; %Matriks Observability Oi pers 3.85
69     Si=pinv(Oi)*Pi; %Kalman Filter State Sequence pers 3.86
70
71     Oim=Uk(1:1*(i-1),:)*ss; %Matriks Obsevability Oi dengan 1 baris terakhir dihapus
72     Sip=pinv(Oim)*Pim; %Kalman State Sequence pers 3.88
73
74     RHS=[Sip;Yi]*pinv(Si);
75     A=RHS(1:k,:); %Matriks State A pers 3.89
76     %Ci=RHS(k+1:k+1,:); %Ouput Matriks Pengaruh C pers 3.89
77
78     [phi,M] = eig(A); %Solusi Eigenvalue matriks A pers 3.90
79     my=diag(M); %Eigenvalue dari Matriks Diagonal M
80
81     %mode(k)=[Ci]*phi; %Bentuk Ragam/Mode shapes pers 3.91
82
83     lambda= log(my)/dt; %Dari Waktu Diskrit ke Waktu Kontinyu
84
85     f(1:k,k)=abs(lambda); %frekuensi alami (rad/s) pers 3.93
86     %fd(1:k,k)=imag(lambda); %Frekuensi Modal Teredam (rad/s) 3.94
87     c(1:k,k)=-real(lambda)./abs(lambda); %rasio redaman pers 3.95
88 end
89 %-----%
90 diagstab(f,c,n,1); %Diagram Stabilisasi
91 end

```

1. frame3lt.m

Editor - D:\tesis\analisis Matlab\frame3lt.m



```

1 - clear all
2 - clc
3
4 - n1=coor(0,0);    %--data koordinat
5 - n2=coor(6,0);
6 - n3=coor(0,4);
7 - n4=coor(6,4);
8 - n5=coor(0,8);
9 - n6=coor(6,8);
10 - n7=coor(0,12);
11 - n8=coor(6,12);
12
13 - E=2.35e7;      %--kN/m^2
14 - rho=2400;     %--kg/m^3
15
16 - A1=0.45*0.45;
17 - I1=1/12*0.45*0.45^3;
18 - [L1,T1]=memf(n1,n3);    %--menghitung L dan T
19 - k1=klf(E,A1,I1,L1);    %--k lokal
20 - K1=kg(k1,T1);         %--K global
21 - ID1=[0 0 0 13 1 2];   %-- vektor tujuan
22
23 - A2=A1;
24 - I2=I1;
25 - [L2,T2]=memf(n2,n4);
26 - k2=klf(E,A2,I2,L2);
27 - K2=kg(k2,T2);
28 - ID2=[0 0 0 13 3 4];
29
30 - A3=A1;
31 - I3=I1;
32 - [L3,T3]=memf(n3,n5);
33 - k3=klf(E,A3,I3,L3);
34 - K3=kg(k3,T3);
35 - ID3=[13 1 2 14 5 6];
36
37 - A4=A1;
38 - I4=I1;
39 - [L4,T4]=memf(n4,n6);
40 - k4=klf(E,A4,I4,L4);
41 - K4=kg(k4,T4);
42 - ID4=[13 3 4 14 7 8];

```

```

43
44 - A5=0.4*0.4;
45 - I5=1/12*0.40*0.40^3;
46 - [L5,T5]=memf(n5,n7);
47 - k5=klf(E,A5,I5,L5);
48 - K5=kg(k5,T5);
49 - ID5=[14 5 6 15 9 10];
50
51 - A6=A5;
52 - I6=I5;
53 - [L6,T6]=memf(n6,n8);
54 - k6=klf(E,A6,I6,L6);
55 - K6=kg(k6,T6);
56 - ID6=[14 7 8 15 11 12];
57
58 - A7=0.3*0.6;
59 - I7=1/12*0.3*0.6^3;
60 - [L7,T7]=memf(n3,n4);
61 - k7=klf(E,A7,I7,L7);
62 - K7=kg(k7,T7);
63 - ID7=[13 1 2 13 3 4];
64
65 - A8=A7;
66 - I8=I7;
67 - [L8,T8]=memf(n5,n6);
68 - k8=klf(E,A8,I8,L8);
69 - K8=kg(k8,T8);
70 - ID8=[14 5 6 14 7 8];
71
72 - A9=A7;
73 - I9=I7;
74 - [L9,T9]=memf(n7,n8);
75 - k9=klf(E,A9,I9,L9);
76 - K9=kg(k9,T9);
77 - ID9=[15 9 10 15 11 12];
78
79 - dof=15;
80
81 - K=assf(K1, ID1, dof);
82 - K=K+assf(K2, ID2, dof);
83 - K=K+assf(K3, ID3, dof);
84 - K=K+assf(K4, ID4, dof);
85 - K=K+assf(K5, ID5, dof);
86 - K=K+assf(K6, ID6, dof);
87 - K=K+assf(K7, ID7, dof);
88 - K=K+assf(K8, ID8, dof);
89 - K=K+assf(K9, ID9, dof); %-- K struktur
90
91 - nc=12;
92 - nl=3;
93 - Klat=kcon(K,nc,nl)
94 - %*****

```

```

95 - n=size(Klat);
96 - n=n(1); %--ukuran DOF atau n=2
97 - N=2*n; %--ukuran state vector
98 - %
99 - %--Matriks massa
100 - M=[25 0 0;0 25 0;0 0 25]
101 - eo=[-diag(M)];
102 -
103 - [eigv,eigval]=eig(M\Klat);
104 - [wo,worder]=sort(sqrt(diag(eigval)));
105 - for i = 1:3
106 -     mode(:,i)=eigv(:,i)/eigv(3,i);
107 - end
108 - mode1=[mode(3,3);mode(2,3);mode(1,3)];
109 - mode2=[mode(3,2);mode(2,2);mode(1,2)];
110 - mode3=[mode(3,1);mode(2,1);mode(1,1)];
111 -
112 - %--matriks redaman c
113 - T1=2*pi/wo(1); %---waktu getar
114 - T2=2*pi/wo(2);
115 - T3=2*pi/wo(3);
116 -
117 - rd=0.02 %--rasio redaman 2%
118 - ak=rd*T1/pi;
119 - am=rd*4*pi/T1;
120 -
121 - Ck=ak*Klat; %--redaman sebanding kekakuan
122 - Cm=am*M; %--redaman sebanding massa
123 - C=Ck+Cm %--redaman Rayleigh
124 -
125 - %--State space Eq
126 - B=[zeros(n,n) eye(n);-inv(M)*Klat -inv(M)*C];
127 - E=[zeros(n,1);inv(M)*eo];
128 -
129 - %--Y=X
130 - Cy=eye(N);
131 - Dy=zeros(N,1);
132 -
133 - syst1=ss(B,E,Cy,Dy);
134 -
135 - %--getaran random
136 - t1=(0:0.01:10^3);
137 - iul=randn(length(t1),1);
138 -
139 - [y1,t1,z1]=lsim(syst1,iul,t1); %%simulasi
140 -
141 - %--percepatan
142 - I0=[1;1;1];
143 - accframe3lt=-[inv(M)*Klat inv(M)*C]*z1'-I0*iul';
144 -
145 - save accframe3lt.mat

```


2.truss12.m

Editor - D:\tesis\analisis Matlab\truss12.m

```

EDITOR      PUBLISH      VIEW
+   New   Open   Save   Find Files   Compare   Print   Go To   Find   Insert   Comment   Indent   Breakpoints   Run   Run and Advance
FILE      NAVIGATE      EDIT      BREAKPOINTS      RUN

truss12.m  x  +
1  %%aplikasi struktur rangka batang
2  clear all
3  clc
4
5  %%--koordinat titik
6  n1=coor(0,0);
7  n2=coor(0.3,0);
8  n3=coor(0.6,0);
9  n4=coor(0.9,0);
10 n5=coor(1.2,0);
11 n6=coor(0.3,0.4);
12 n7=coor(0.6,0.4);
13 n8=coor(0.9,0.4);
14
15 %%input
16 E= 2e8;           % Modulus Elastisitas (kN/m^2)
17 D=0.006;         % diameter batang (m)
18 A= 0.25*pi*D^2;  % Luas tampang batang (m2)
19 rho=1.18/5;      % berat per meter panjang (kg/m')
20
21 %%batang 1
22 [L1,T1]=memt(n1,n2);    %%--menghitung L dan T
23 k1=klt(E,A,L1);        %%--kekakuan lokal
24 K1=kg(k1,T1);          %%--kekakuan global
25 m1=mlt(rho,L1);        %%--massa lokal
26 M1=mg(m1,T1);          %%--massa global
27 id1=[0 0 1 2];         %%--vektor tujuan
28
29 %%batang 2
30 [L2,T2]=memt(n2,n3);    %%--menghitung L dan T
31 k2=klt(E,A,L2);        %%--kekakuan lokal
32 K2=kg(k2,T2);          %%--kekakuan global
33 m2=mlt(rho,L2);        %%--massa lokal
34 M2=mg(m2,T2);          %%--massa global
35 id2=[1 2 3 4];         %%--vektor tujuan
36
37 %%batang 3
38 [L3,T3]=memt(n3,n4);    %%--menghitung L dan T
39 k3=klt(E,A,L3);        %%--kekakuan lokal
40 K3=kg(k3,T3);          %%--kekakuan global
41 m3=mlt(rho,L3);        %%--massa lokal
42 M3=mg(m3,T3);          %%--massa global
43 id3=[3 4 5 6];         %%--vektor tujuan

```

```

44
45 %%batang 4
46 - [L4,T4]=memt(n4,n5); %%--menghitung L dan T
47 - k4=klt(E,A,L4); %%--kekakuan lokal
48 - K4=kg(k4,T4); %%--kekakuan global
49 - m4=mlt(rho,L4); %%--massa lokal
50 - M4=mg(m4,T4); %%--massa global
51 - id4=[5 6 0 0]; %%--vektor tujuan
52
53 %%batang 5
54 - [L5,T5]=memt(n6,n7); %%--menghitung L dan T
55 - k5=klt(E,A,L5); %%--kekakuan lokal
56 - K5=kg(k5,T5); %%--kekakuan global
57 - m5=mlt(rho,L5); %%--massa lokal
58 - M5=mg(m5,T5); %%--massa global
59 - id5=[7 8 9 10]; %%--vektor tujuan
60
61 %%batang 6
62 - [L6,T6]=memt(n7,n8); %%--menghitung L dan T
63 - k6=klt(E,A,L6); %%--kekakuan lokal
64 - K6=kg(k6,T6); %%--kekakuan global
65 - m6=mlt(rho,L6); %%--massa lokal
66 - M6=mg(m6,T6); %%--massa global
67 - id6=[9 10 11 12]; %%--vektor tujuan
68
69 %%batang 7
70 - [L7,T7]=memt(n2,n6); %%--menghitung L dan T
71 - k7=klt(E,A,L7); %%--kekakuan lokal
72 - K7=kg(k7,T7); %%--kekakuan global
73 - m7=mlt(rho,L7); %%--massa lokal
74 - M7=mg(m7,T7); %%--massa global
75 - id7=[1 2 7 8]; %%--vektor tujuan
76
77 %%batang 8
78 - [L8,T8]=memt(n3,n7); %%--menghitung L dan T
79 - k8=klt(E,A,L8); %%--kekakuan lokal
80 - K8=kg(k8,T8); %%--kekakuan global
81 - m8=mlt(rho,L8); %%--massa lokal
82 - M8=mg(m8,T8); %%--massa global
83 - id8=[3 4 9 10]; %%--vektor tujuan
84
85 %%batang 9
86 - [L9,T9]=memt(n4,n8); %%--menghitung L dan T
87 - k9=klt(E,A,L9); %%--kekakuan lokal
88 - K9=kg(k9,T9); %%--kekakuan global
89 - m9=mlt(rho,L9); %%--massa lokal
90 - M9=mg(m9,T9); %%--massa global
91 - id9=[5 6 11 12]; %%--vektor tujuan
92
93 %%batang 10
94 - [L10,T10]=memt(n1,n6); %%--menghitung L dan T
95 - k10=klt(E,A,L10); %%--kekakuan lokal
96 - K10=kg(k10,T10); %%--kekakuan global
97 - m10=mlt(rho,L10); %%--massa lokal
98 - M10=mg(m10,T10); %%--massa global
99 - id10=[0 0 7 8]; %%--vektor tujuan
100
101 %%batang 11
102 - [L11,T11]=memt(n6,n3); %%--menghitung L dan T
103 - k11=klt(E,A,L11); %%--kekakuan lokal

```

```

104 - K11=kg(k11,T11); %--kekakuan global
105 - m11=mlt(rho,L11); %--massa lokal
106 - M11=mg(m11,T11); %--massa global
107 - id11=[7 8 3 4]; %--vektor tujuan
108
109 %%batang 12
110 - [L12,T12]=memt(n3,n8); %--menghitung L dan T
111 - k12=klt(E,A,L12); %--kekakuan lokal
112 - K12=kg(k12,T12); %--kekakuan global
113 - m12=mlt(rho,L12); %--massa lokal
114 - M12=mg(m12,T12); %--massa global
115 - id12=[3 4 11 12]; %--vektor tujuan
116
117 %%batang 13
118 - [L13,T13]=memt(n8,n5); %--menghitung L dan T
119 - k13=klt(E,A,L13); %--kekakuan lokal
120 - K13=kg(k13,T13); %--kekakuan global
121 - m13=mlt(rho,L13); %--massa lokal
122 - M13=mg(m13,T13); %--massa global
123 - id13=[11 12 0 0]; %--vektor tujuan
124
125 - dof=12;
126
127 - K=asst(K1,id1,dof);
128 - K=K+asst(K2,id2,dof);
129 - K=K+asst(K3,id3,dof);
130 - K=K+asst(K4,id4,dof);
131 - K=K+asst(K5,id5,dof);
132 - K=K+asst(K6,id6,dof);
133 - K=K+asst(K7,id7,dof);
134 - K=K+asst(K8,id8,dof);
135 - K=K+asst(K9,id9,dof);
136 - K=K+asst(K10,id10,dof);
137 - K=K+asst(K11,id11,dof);
138 - K=K+asst(K12,id12,dof);
139 - K=K+asst(K13,id13,dof); %--Kekakuan struktur
140
141 - M=asst(M1,id1,dof);
142 - M=M+asst(M2,id2,dof);
143 - M=M+asst(M3,id3,dof);
144 - M=M+asst(M4,id4,dof);
145 - M=M+asst(M5,id5,dof);
146 - M=M+asst(M6,id6,dof);
147 - M=M+asst(M7,id7,dof);
148 - M=M+asst(M8,id8,dof);
149 - M=M+asst(M9,id9,dof);
150 - M=M+asst(M10,id10,dof);
151 - M=M+asst(M11,id11,dof);
152 - M=M+asst(M12,id12,dof);
153 - M=M+asst(M13,id13,dof); %-- Massa struktur
154 - %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
155
156 - %nc=6
157 - %nv=6
158 - %Kver=kcon(K,nc,nv) %--Kondensasi matriks kekakuan
159 - %Mver=kcon(M,nc,nv) %--Kondensasi matriks massa
160 - %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
161
162 - n=size(K);
163 - n=n(1); %--ukuran DOF atau n=2

```

```

164 - N=2*n;    %--ukuran state vector
165
166 - eo=[-diag(M)];
167
168 - [eigv,eigval]=eig(M\K);
169 - [wo,worder]=sort(sqrt(diag(eigval)))
170 - modeshape=eigv(:,worder);
171 - for i=1:12
172 -     modes(:,i)=modeshape(:,i)/modeshape(12,i);
173 - end
174
175     %--matriks redaman c
176 - Tl=2*pi/wo(1);          %---waktu getar
177
178 - rd=0.02                %--rasio redaman 2%
179 - ak=rd*Tl/pi;
180 - am=rd*4*pi/Tl;
181
182 - C=(ak*K)+(am*M);      %--redaman rayleigh
183 %C=am*M
184 %*****
185
186 %--Persamaan State space
187 - A=[zeros(n,n) eye(n);-inv(M)*K -inv(M)*C];
188 - E=[zeros(n,1);inv(M)*eo];
189
190 %--Y=X
191 - Cy=eye(N);
192 - Dy=zeros(N,1);
193
194 - syst1=ss(A,E,Cy,Dy);
195
196 %--getaran random
197 - t1=(0.01:0.01:10^3);
198 - iul=randn(length(t1),1);
199
200 - [y1,t1,z1]=lsim(syst1,iul,t1);    %%simulasi
201 %plot(t1,y1)
202 %plot(t1,y1(:,1),'-k')    %% plot perpindahan lantai 1 dan 2
203 %xlabel('waktu (detik)')
204 %ylabel('Perpindahan (m)')
205
206 %y1max=(max(abs(y1(:,1))))
207 %y2max=(max(abs(y1(:,2))))    %%%---perpindahan max
208
209 - I0=[1;1;1;1;1;1;1;1;1;1;1;1];
210 - acctruss12=-[inv(M)*K inv(M)*C]*z1'-I0*iul';
211     %-[inv(M)*Klat inv(M)*C]*z1'-I0*iul';
212
213 %percepatan_Lt1=acc2lt(1,:);
214 - save acctruss12.mat
215
216 %fix=acc2lt'

```


3.balokpazz.m

Editor - D:\tesis\analisis Matlab\balokpazz.m

```

EDITOR      PUBLISH      VIEW
+   New   Open   Save   Find Files   Compare   Go To   Insert   Comment   Indent   Breakpoints   Run   Run and Advance
FILE      NAVIGATE      EDIT      BREAKPOINTS      RUN

balokpazz.m  x  +
1  %Analisis eksak untuk konfigurasi struktur balok beton bertulang
2  clear all
3  clc
4
5  %%--koordinat titik
6  n1=coor(0,0);
7  n2=coor(1,0);
8  n3=coor(2,0);
9  n4=coor(3,0);
10 n5=coor(4,0);
11
12 %%--input
13 A=0.25*0.5;
14 I=(1/12)*0.25*0.5^3;      %%--momen inersia (m4)
15 E=2.35e7;                %%--modulus elastisitas (KN/m2)
16 rho=2400;                %%--massa jenis (kg/m3)
17
18 %%batang 1
19 [L1,T1]=memb(n1,n2);      %%--menghitung L dan T
20 k1=k1b(E,I,L1);          %%--kekakuan lokal
21 K1=kg(k1,T1);            %%--kekakuan global
22 m1=mlb(rho,A,L1);        %%--massa lokal
23 M1=mg(m1,T1);           %%--massa global
24 id1=[0 0 4 1];          %%--vektor tujuan
25
26 %%batang 2
27 [L2,T2]=memb(n2,n3);      %%--menghitung L dan T
28 k2=k1b(E,I,L2);          %%--kekakuan lokal
29 K2=kg(k2,T2);            %%--kekakuan global
30 m2=mlb(rho,A,L2);        %%--massa lokal
31 M2=mg(m2,T2);           %%--massa global
32 id2=[4 1 5 2];          %%--vektor tujuan
33
34 %%batang 3
35 [L3,T3]=memb(n3,n4);      %%--menghitung L dan T
36 k3=k1b(E,I,L3);          %%--kekakuan lokal
37 K3=kg(k3,T3);            %%--kekakuan global
38 m3=mlb(rho,A,L3);        %%--massa lokal
39 M3=mg(m3,T3);           %%--massa global
40 id3=[5 2 6 3];          %%--vektor tujuan
41

```

```

42 %%batang 4
43 - [L4,T4]=memb(n4,n5); %%--menghitung L dan T
44 - k4=klb(E,I,L4); %%--kekakuan lokal
45 - K4=kg(k4,T4); %%--kekakuan global
46 - m4=mlb(rho,A,L4); %%--massa lokal
47 - M4=mg(m4,T4); %%--massa global
48 - id4=[6 3 0 0]; %%--vektor tujuan
49
50 - dof=6;
51
52 - K=assb(K1,id1,dof);
53 - K=K+assb(K2,id2,dof);
54 - K=K+assb(K3,id3,dof);
55 - K=K+assb(K4,id4,dof); %%--K struktur
56
57 - M=assb(M1,id1,dof);
58 - M=M+assb(M2,id2,dof);
59 - M=M+assb(M3,id3,dof);
60 - M=M+assb(M4,id4,dof); %%--M struktur
61
62 - nc=3;
63 - nv=3;
64 - Kver=kcon(K,nc,nv) %%--Kondensasi matriks kekakuan
65 - Mver=kcon(M,nc,nv) %%--Kondensasi matriks massa
66
67 - n=size(Kver);
68 - n=n(1); %%--ukuran DOF atau n=2
69 - N=2*n; %%--ukuran state vector
70
71 - eo=[-diag(Mver)];

```

```

72
73 - [eigv, eigval]=eig(Mver\Kver);
74 - [wo,worder]=sort(sqrt(diag(eigval)));
75 - for i = 1:3
76 -     mode(:,i)=eigv(:,i)/eigv(1,i);
77 - end
78 - mode1=[mode(3,3);mode(2,3);mode(1,3)];
79 - mode2=[mode(3,2);mode(2,2);mode(1,2)];
80 - mode3=[mode(3,1);mode(2,1);mode(1,1)];
81
82 %--matriks redaman c
83 - T1=2*pi/wo(1); %---waktu getar
84
85 - rd=0.02 %--rasio redaman 2%
86 - ak=rd*T1/pi;
87 - am=rd*4*pi/T1;
88
89 - C=(ak*Kver)+(am*Mver); %--redaman rayleigh
90
91 %--State space Eq
92 - B=[zeros(n,n) eye(n);-inv(Mver)*Kver -inv(Mver)*C];
93 - E=[zeros(n,1);inv(Mver)*eo];
94
95 %--Y=X
96 - Cy=eye(N);
97 - Dy=zeros(N,1);
98
99 - syst1=ss(B,E,Cy,Dy);
100
101 %--getaran random
102 - t1=(0:0.01:10^3);
103 - iul=randn(length(t1),1);
104
105 - [y1,t1,z1]=lsim(syst1,iul,t1); %%simulasi
106
107 - I0=[1;1;1];
108 - accbalokp=-[inv(Mver)*Kver inv(Mver)*C]*z1'-I0*iul';
109
110 - save accbalokp.mat

```

4. frame3ltrusak.m

```

Editor - D:\tesis\analisis Matlab\frame3ltrusak.m
EDITOR PUBLISH VIEW
+ Find Files Insert fx fi
New Open Save Compare Go To Comment % % %
Print Find Indent Breakpoints Run Run and Run
FILE NAVIGATE EDIT BREAKPOINTS RUN
frame3ltrusak.m x balokpazz.m x +
1 %Analisis eksak untuk rangka geser rusak
2 - clear all
3 - clc
4
5 - n1=coor(0,0); %--data koordinat
6 - n2=coor(6,0);
7 - n3=coor(0,4);
8 - n4=coor(6,4);
9 - n5=coor(0,8);
10 - n6=coor(6,8);
11 - n7=coor(0,12);
12 - n8=coor(6,12);
13
14 - E=2.35e7; %--kN/m^2
15 - rho=2400; %--kg/m^3
16
17 - A1=0.2*0.2;
18 - I1=1/12*0.25*0.25;
19 - [L1,T1]=memf(n1,n3); %--menghitung L dan T
20 - k1=k1f(E,A1,I1,L1); %--k lokal
21 - K1=kg(k1,T1); %--K global
22 - ID1=[0 0 0 13 1 2]; %-- vektor tujuan
23
24 - A2=A1;
25 - I2=I1;
26 - [L2,T2]=memf(n2,n4);
27 - k2=k1f(E,A2,I2,L2);
28 - K2=kg(k2,T2);
29 - ID2=[0 0 0 13 3 4];
30
31 - A3=A1;
32 - I3=I1;
33 - [L3,T3]=memf(n3,n5);
34 - k3=k1f(E,A3,I3,L3);
35 - K3=kg(k3,T3);
36 - ID3=[13 1 2 14 5 6];
37
38 - A4=A1;
39 - I4=I1;
40 - [L4,T4]=memf(n4,n6);
41 - k4=k1f(E,A4,I4,L4);
42 - K4=kg(k4,T4);
43 - ID4=[13 3 4 14 7 8];
44
45 - A5=0.2*0.2;
46 - I5=1/12*0.2*0.2^3;
47 - [L5,T5]=memf(n5,n7);
48 - k5=k1f(E,A5,I5,L5);
49 - K5=kg(k5,T5);
50 - ID5=[14 5 6 15 9 10];

```



```

52 - A6=A5;
53 - I6=I5;
54 - [L6,T6]=memf(n6,n8);
55 - k6=klf(E,A6,I6,L6);
56 - K6=kg(k6,T6);
57 - ID6=[14 7 8 15 11 12];
58
59 - A7=0.15*0.3;
60 - I7=1/12*0.15*0.3^3;
61 - [L7,T7]=memf(n3,n4);
62 - k7=klf(E,A7,I7,L7);
63 - K7=kg(k7,T7);
64 - ID7=[13 1 2 13 3 4];
65
66 - A8=A7;
67 - I8=I7;
68 - [L8,T8]=memf(n5,n6);
69 - k8=klf(E,A8,I8,L8);
70 - K8=kg(k8,T8);
71 - ID8=[14 5 6 14 7 8];
72
73 - A9=A7;
74 - I9=I7;
75 - [L9,T9]=memf(n7,n8);
76 - k9=klf(E,A9,I9,L9);
77 - K9=kg(k9,T9);
78 - ID9=[15 9 10 15 11 12];
79
80 - dof=15;
81
82 - K=assf(K1, ID1, dof);
83 - K=K+assf(K2, ID2, dof);
84 - K=K+assf(K3, ID3, dof);
85 - K=K+assf(K4, ID4, dof);
86 - K=K+assf(K5, ID5, dof);
87 - K=K+assf(K6, ID6, dof);
88 - K=K+assf(K7, ID7, dof);
89 - K=K+assf(K8, ID8, dof);
90 - K=K+assf(K9, ID9, dof);    %-- K struktur
91
92 - nc=12;
93 - nl=3;
94 - Klat=kcon(K,nc,nl)
95 - %%%%%%%%%%%
96 - n=size(Klat);
97 - n=n(1);    %--ukuran DOF atau n=2
98 - N=2*n;    %--ukuran state vector
99 - %
100 - %--Matriks massa
101 - M=[25 0 0;0 25 0;0 0 25]
102 - eo=[-diag(M)];
103
104 - [eigv,eigval]=eig(M\Klat);
105 - [wo,worder]=sort(sqrt(diag(eigval)))
106 - for i = 1:3
107 -     mode(:,i)=eigv(:,i)/eigv(3,i);
108 - end
109 - mode1=[mode(3,3);mode(2,3);mode(1,3)];
110 - mode2=[mode(3,2);mode(2,2);mode(1,2)];
111 - mode3=[mode(3,1);mode(2,1);mode(1,1)];

```

```

112
113 %--matriks redaman c
114 - T1=2*pi/wo(1); %---waktu getar
115 - T2=2*pi/wo(2);
116 - T3=2*pi/wo(3);
117
118 - rd=0.02 %--rasio redaman 2%
119 - ak=rd*T1/pi;
120 - am=rd*4*pi/T1;
121
122 - Ck=ak*Klat; %--redaman sebanding kekakuan
123 - Cm=am*M; %--redaman sebanding massa
124 - C=Ck+Cm ; %--redaman Rayleigh
125
126 %--State space Eq
127 - B=[zeros(n,n) eye(n);-inv(M)*Klat -inv(M)*C];
128 - E=[zeros(n,1);inv(M)*eo];
129
130 %--Y=X
131 - Cy=eye(N);
132 - Dy=zeros(N,1);
133
134 - syst1=ss(B,E,Cy,Dy);
135
136 %--getaran random
137 - t1=(0:0.01:10^3);
138 - iul=randn(length(t1),1);
139
140 - [y1,t1,z1]=lsim(syst1,iul,t1); %%simulasi
141
142 %--percepatan
143 - I0=[1;1;1];
144 - accframe3ltrusak=-[inv(M)*Klat inv(M)*C]*z1'-I0*iul';
145
146 - save accframe3ltrusak.mat

```

5. truss12rusak.m

```

Editor - D:\tesis\analisis Matlab\truss12rusak.m

EDITOR PUBLISH VIEW
+ New Open Save Find Files Compare Go To Insert Comment Indent Breakpoints Run Run and Advance
FILE NAVIGATE EDIT BREAKPOINTS RUN

truss12rusak.m x +
1 %%aplikasi struktur rangka batang rusak
2 clear all
3 clc
4
5 %%--koordinat titik
6 n1=coor(0,0);
7 n2=coor(0.3,0);
8 n3=coor(0.6,0);
9 n4=coor(0.9,0);
10 n5=coor(1.2,0);
11 n6=coor(0.3,0.4);
12 n7=coor(0.6,0.4);
13 n8=coor(0.9,0.4);
14
15 %%input
16 E= 2e8; % Modulus Elastisitas (kN/m^2)
17 D=0.004; % diameter batang (m)
18 A= 0.25*pi*D^2; % Luas tampang batang (m2)
19 rho=1.18/5; % berat per meter panjang (kg/m')
20
21 %%batang 1
22 [L1,T1]=memt(n1,n2); %--menghitung L dan T
23 k1=klt(E,A,L1); %--kekakuan lokal
24 K1=kg(k1,T1); %--kekakuan global
25 m1=mlt(rho,L1); %--massa lokal
26 M1=mg(m1,T1); %--massa global
27 id1=[0 0 1 2]; %--vektor tujuan
28
29 %%batang 2
30 [L2,T2]=memt(n2,n3); %--menghitung L dan T
31 k2=klt(E,A,L2); %--kekakuan lokal
32 K2=kg(k2,T2); %--kekakuan global
33 m2=mlt(rho,L2); %--massa lokal
34 M2=mg(m2,T2); %--massa global
35 id2=[1 2 3 4]; %--vektor tujuan
36
37 %%batang 3
38 [L3,T3]=memt(n3,n4); %--menghitung L dan T
39 k3=klt(E,A,L3); %--kekakuan lokal
40 K3=kg(k3,T3); %--kekakuan global
41 m3=mlt(rho,L3); %--massa lokal
42 M3=mg(m3,T3); %--massa global
43 id3=[3 4 5 6]; %--vektor tujuan
44
45 %%batang 4
46 [L4,T4]=memt(n4,n5); %--menghitung L dan T
47 k4=klt(E,A,L4); %--kekakuan lokal
48 K4=kg(k4,T4); %--kekakuan global
49 m4=mlt(rho,L4); %--massa lokal
50 M4=mg(m4,T4); %--massa global
51 id4=[5 6 0 0]; %--vektor tujuan
52

```

```

53 %%batang 5
54 - [L5,T5]=memt(n6,n7); %%--menghitung L dan T
55 - k5=klt(E,A,L5); %%--kekakuan lokal
56 - K5=kg(k5,T5); %%--kekakuan global
57 - m5=mlt(rho,L5); %%--massa lokal
58 - M5=mg(m5,T5); %%--massa global
59 - id5=[7 8 9 10]; %%--vektor tujuan
60
61 %%batang 6
62 - [L6,T6]=memt(n7,n8); %%--menghitung L dan T
63 - k6=klt(E,A,L6); %%--kekakuan lokal
64 - K6=kg(k6,T6); %%--kekakuan global
65 - m6=mlt(rho,L6); %%--massa lokal
66 - M6=mg(m6,T6); %%--massa global
67 - id6=[9 10 11 12]; %%--vektor tujuan
68
69 %%batang 7
70 - [L7,T7]=memt(n2,n6); %%--menghitung L dan T
71 - k7=klt(E,A,L7); %%--kekakuan lokal
72 - K7=kg(k7,T7); %%--kekakuan global
73 - m7=mlt(rho,L7); %%--massa lokal
74 - M7=mg(m7,T7); %%--massa global
75 - id7=[1 2 7 8]; %%--vektor tujuan
76
77 %%batang 8
78 - [L8,T8]=memt(n3,n7); %%--menghitung L dan T
79 - k8=klt(E,A,L8); %%--kekakuan lokal
80 - K8=kg(k8,T8); %%--kekakuan global
81 - m8=mlt(rho,L8); %%--massa lokal
82 - M8=mg(m8,T8); %%--massa global
83 - id8=[3 4 9 10]; %%--vektor tujuan
84
85 %%batang 9
86 - [L9,T9]=memt(n4,n8); %%--menghitung L dan T
87 - k9=klt(E,A,L9); %%--kekakuan lokal
88 - K9=kg(k9,T9); %%--kekakuan global
89 - m9=mlt(rho,L9); %%--massa lokal
90 - M9=mg(m9,T9); %%--massa global
91 - id9=[5 6 11 12]; %%--vektor tujuan
92
93 %%batang 10
94 - [L10,T10]=memt(n1,n6); %%--menghitung L dan T
95 - k10=klt(E,A,L10); %%--kekakuan lokal
96 - K10=kg(k10,T10); %%--kekakuan global
97 - m10=mlt(rho,L10); %%--massa lokal
98 - M10=mg(m10,T10); %%--massa global
99 - id10=[0 0 7 8]; %%--vektor tujuan
100

```



```

101 %%batang 11
102 - [L11,T11]=memt(n6,n3); %%--menghitung L dan T
103 - k11=klt(E,A,L11); %%--kekakuan lokal
104 - K11=kg(k11,T11); %%--kekakuan global
105 - m11=mlt(rho,L11); %%--massa lokal
106 - M11=mg(m11,T11); %%--massa global
107 - id11=[7 8 3 4]; %%--vektor tujuan
108
109 %%batang 12
110 - [L12,T12]=memt(n3,n8); %%--menghitung L dan T
111 - k12=klt(E,A,L12); %%--kekakuan lokal
112 - K12=kg(k12,T12); %%--kekakuan global
113 - m12=mlt(rho,L12); %%--massa lokal
114 - M12=mg(m12,T12); %%--massa global
115 - id12=[3 4 11 12]; %%--vektor tujuan
116
117 %%batang 13
118 - [L13,T13]=memt(n8,n5); %%--menghitung L dan T
119 - k13=klt(E,A,L13); %%--kekakuan lokal
120 - K13=kg(k13,T13); %%--kekakuan global
121 - m13=mlt(rho,L13); %%--massa lokal
122 - M13=mg(m13,T13); %%--massa global
123 - id13=[11 12 0 0]; %%--vektor tujuan
124
125 - dof=12;
126
127 - K=asst(K1,id1,dof);
128 - K=K+asst(K2,id2,dof);
129 - K=K+asst(K3,id3,dof);
130 - K=K+asst(K4,id4,dof);
131 - K=K+asst(K5,id5,dof);
132 - K=K+asst(K6,id6,dof);
133 - K=K+asst(K7,id7,dof);
134 - K=K+asst(K8,id8,dof);
135 - K=K+asst(K9,id9,dof);
136 - K=K+asst(K10,id10,dof);
137 - K=K+asst(K11,id11,dof);
138 - K=K+asst(K12,id12,dof);
139 - K=K+asst(K13,id13,dof); %%--Kekakuan struktur
140
141 - M=asst(M1,id1,dof);
142 - M=M+asst(M2,id2,dof);
143 - M=M+asst(M3,id3,dof);
144 - M=M+asst(M4,id4,dof);
145 - M=M+asst(M5,id5,dof);
146 - M=M+asst(M6,id6,dof);
147 - M=M+asst(M7,id7,dof);
148 - M=M+asst(M8,id8,dof);
149 - M=M+asst(M9,id9,dof);
150 - M=M+asst(M10,id10,dof);
151 - M=M+asst(M11,id11,dof);
152 - M=M+asst(M12,id12,dof);
153 - M=M+asst(M13,id13,dof); %%-- Massa struktur
154 - %%%%%%%%%%%
155

```

```

162 - n=size(K);
163 - n=n(1); %--ukuran DOF atau n=2
164 - N=2*n; %--ukuran state vector
165
166 - eo=[-diag(M)];
167
168 - [eigv,eigval]=eig(M\K);
169 - [wo,worder]=sort(sqrt(diag(eigval)));
170 - modeshape=eigv(:,worder);
171 - for i=1:12
172 -     modes(:,i)=modeshape(:,i)/modeshape(12,i);
173 - end
174
175 - %--matriks redaman c
176 - Tl=2*pi/wo(1) %---waktu getar
177
178 - rd=0.02 %--rasio redaman 2%
179 - ak=rd*Tl/pi;
180 - am=rd*4*pi/Tl;
181
182 - C=(ak*K)+(am*M); %--redaman rayleigh
183 - %%%%%%%%%%%
184
185 - %--Persamaan State space
186 - A=[zeros(n,n) eye(n);-inv(M)*K -inv(M)*C];
187 - E=[zeros(n,1);inv(M)*eo];
188
189 - %--Y=X
190 - Cy=eye(N);
191 - Dy=zeros(N,1);
192
193 - systl=ss(A,E,Cy,Dy);
194
195 - %--getaran random
196 - t1=(0.01:0.01:10^3);
197 - iul=randn(length(t1),1);
198
199 - [y1,t1,z1]=lsim(systl,iul,t1); %%simulasi
200
201 - I0=[1;1;1;1;1;1;1;1;1;1;1];
202 - acctrussl2rusak=-[inv(M)*K inv(M)*C]*z1'-I0*iul';
203
204 - save acctrussl2rusak.mat

```

5. balokpazrusak.m

```

Editor - D:\tesis\analisis Matlab\balokpazrusak.m
EDITOR PUBLISH VIEW
+ New Open Save Find Files Compare Print Go To Find Comment Indent Insert fx % % % Breakpoints Run Run and Advance Adv
FILE NAVIGATE EDIT BREAKPOINTS RUN

balokpazrusak.m x +
1 %Analisis eksak untuk konfigurasi struktur balok beton bertulang rusak
2 clear all
3 clc
4
5 %%--koordinat titik
6 n1=coor(0,0);
7 n2=coor(1,0);
8 n3=coor(2,0);
9 n4=coor(3,0);
10 n5=coor(4,0);
11
12 %%--input
13 A=0.1*0.25;
14 I=(1/12)*0.1*0.25^3; %%--momen inersia (m4)
15 E=2.35e7; %%--modulus elastisitas (KN/m2)
16 rho=2400; %%--massa jenis (kg/m3)
17
18 %%batang 1
19 [L1,T1]=memb(n1,n2); %%--menghitung L dan T
20 k1=k1b(E,I,L1); %%--kekakuan lokal
21 K1=kg(k1,T1); %%--kekakuan global
22 m1=mlb(rho,A,L1); %%--massa lokal
23 M1=mg(m1,T1); %%--massa global
24 id1=[0 0 4 1]; %%--vektor tujuan
25
26 %%batang 2
27 [L2,T2]=memb(n2,n3); %%--menghitung L dan T
28 k2=k1b(E,I,L2); %%--kekakuan lokal
29 K2=kg(k2,T2); %%--kekakuan global
30 m2=mlb(rho,A,L2); %%--massa lokal
31 M2=mg(m2,T2); %%--massa global
32 id2=[4 1 5 2]; %%--vektor tujuan
33
34 %%batang 3
35 [L3,T3]=memb(n3,n4); %%--menghitung L dan T
36 k3=k1b(E,I,L3); %%--kekakuan lokal
37 K3=kg(k3,T3); %%--kekakuan global
38 m3=mlb(rho,A,L3); %%--massa lokal
39 M3=mg(m3,T3); %%--massa global
40 id3=[5 2 6 3]; %%--vektor tujuan
41
42 %%batang 4
43 [L4,T4]=memb(n4,n5); %%--menghitung L dan T
44 k4=k1b(E,I,L4); %%--kekakuan lokal
45 K4=kg(k4,T4); %%--kekakuan global
46 m4=mlb(rho,A,L4); %%--massa lokal
47 M4=mg(m4,T4); %%--massa global
48 id4=[6 3 0 0]; %%--vektor tujuan
49
50 dof=6;
51
52 K=assb(K1,id1,dof);
53 K=K+assb(K2,id2,dof);
54 K=K+assb(K3,id3,dof);
55 K=K+assb(K4,id4,dof); %%--K struktur

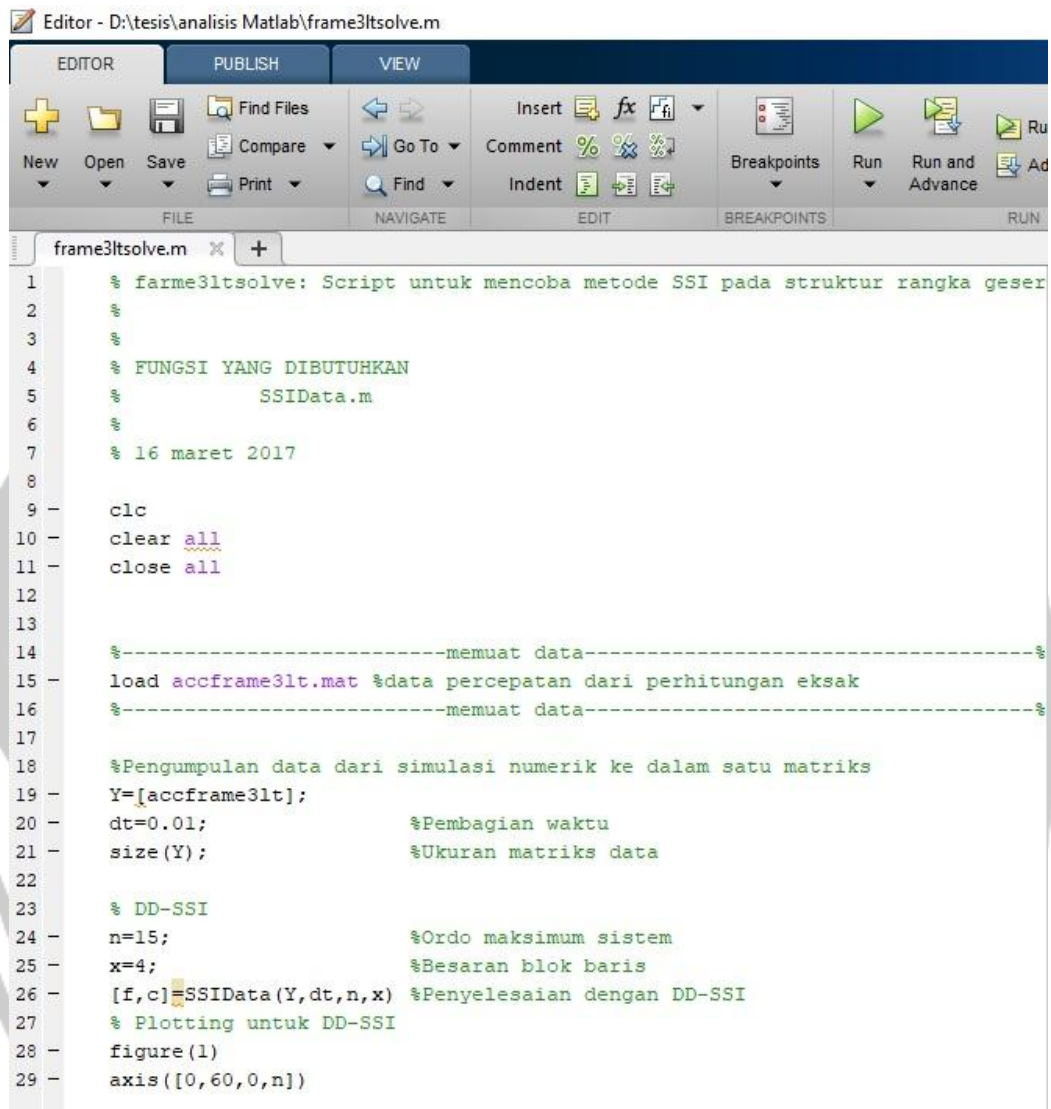
```

```

56
57 - M=assb(M1, id1, dof);
58 - M=M+assb(M2, id2, dof);
59 - M=M+assb(M3, id3, dof);
60 - M=M+assb(M4, id4, dof);           %--M struktur
61
62 - nc=3;
63 - nv=3;
64 - Kver=kcon(K, nc, nv)           %--Kondensasi matriks kekakuan
65 - Mver=kcon(M, nc, nv)           %--Kondensasi matriks massa
66
67
68 - n=size(Kver);
69 - n=n(1); %--ukuran DOF atau n=2
70 - N=2*n; %--ukuran state vector
71
72 - eo=[-diag(Mver)];
73
74 - [eigv, eigval]=eig(Mver\Kver);
75 - [wo, worder]=sort(sqrt(diag(eigval)));
76 - for i = 1:3
77 -     mode(:, i)=eigv(:, i)/eigv(1, i);
78 - end
79 - mode1=[mode(3, 3); mode(2, 3); mode(1, 3)];
80 - mode2=[mode(3, 2); mode(2, 2); mode(1, 2)];
81 - mode3=[mode(3, 1); mode(2, 1); mode(1, 1)];
82
83 %--matriks redaman c
84 - Tl=2*pi/wo(1); %---waktu getar
85
86 - rd=0.02 %--rasio redaman 2%
87 - ak=rd*Tl/pi;
88 - am=rd*4*pi/Tl;
89
90 - C=(ak*Kver)+(am*Mver); %--redaman rayleigh
91
92 %--State space Eq
93 - B=[zeros(n, n) eye(n); -inv(Mver)*Kver -inv(Mver)*C];
94 - E=[zeros(n, 1); inv(Mver)*eo];
95
96 %--Y=X
97 - Cy=eye(N);
98 - Dy=zeros(N, 1);
99
100 - syst1=ss(B, E, Cy, Dy);
101
102 %--getaran random
103 - t1=(0:0.01:10^3);
104 - iul=randn(length(t1), 1);
105
106 - [y1, t1, z1]=lsim(syst1, iul, t1); %%simulasi
107
108 - IO=[1; 1; 1];
109 - accbalokprusak=-[inv(Mver)*Kver inv(Mver)*C]*z1'-IO*iul';
110
111 - save accbalokprusak.mat

```


1.frame3ltsolve.m

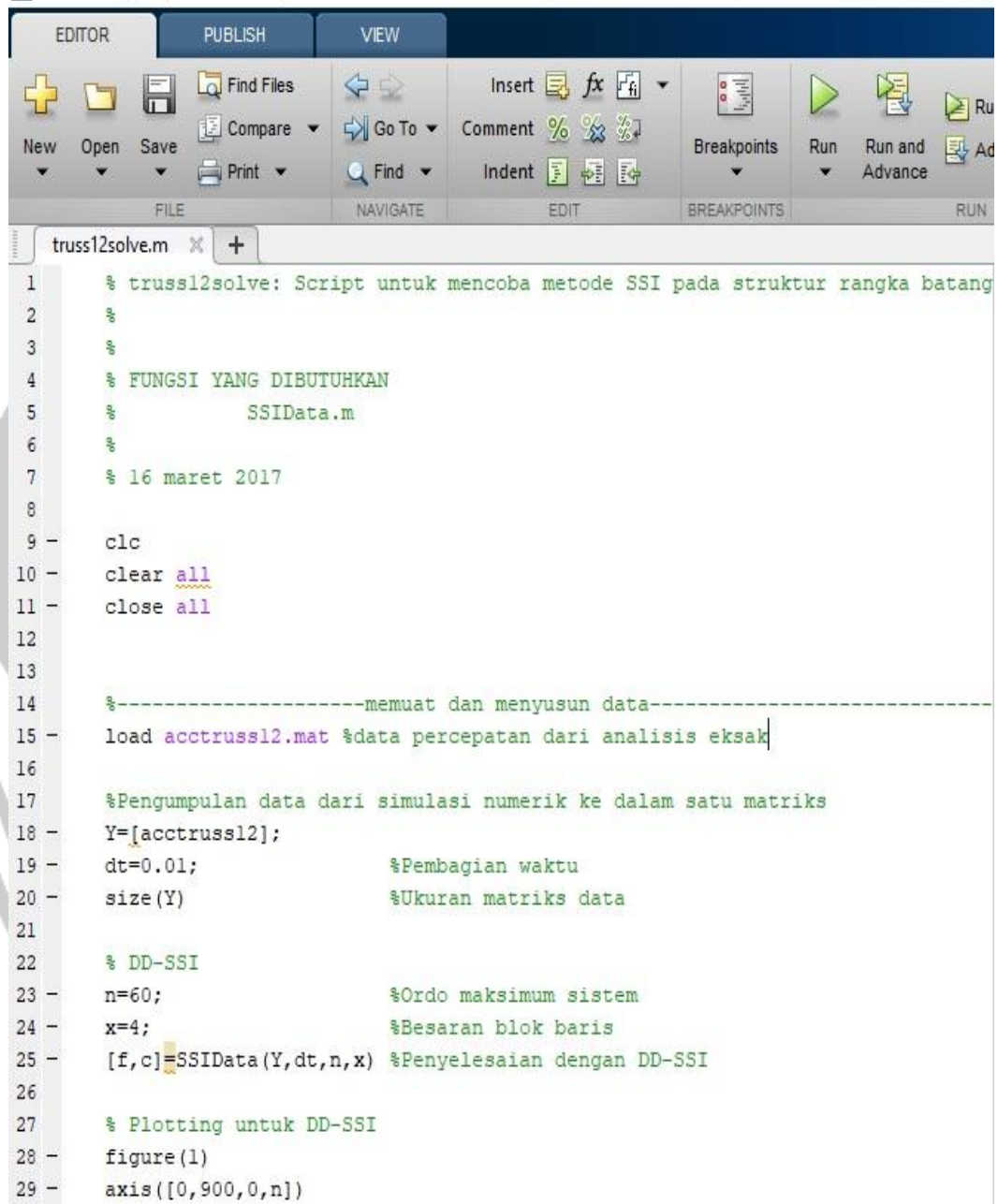


The image shows a screenshot of a MATLAB editor window titled "Editor - D:\tesis\analisis Matlab\frame3ltsolve.m". The window displays the code for a script named "frame3ltsolve.m". The code is as follows:

```
1 % farme3ltsolve: Script untuk mencoba metode SSI pada struktur rangka geser
2 %
3 %
4 % FUNGSI YANG DIBUTUHKAN
5 %     SSIData.m
6 %
7 % 16 maret 2017
8
9 - clc
10 - clear all
11 - close all
12
13
14 %-----memuat data-----%
15 - load accframe3lt.mat %data percepatan dari perhitungan eksak
16 %-----memuat data-----%
17
18 %Pengumpulan data dari simulasi numerik ke dalam satu matriks
19 - Y=[accframe3lt];
20 - dt=0.01; %Pembagian waktu
21 - size(Y); %Ukuran matriks data
22
23 % DD-SSI
24 - n=15; %Ordo maksimum sistem
25 - x=4; %Besaran blok baris
26 - [f,c]=SSIData(Y,dt,n,x) %Penyelesaian dengan DD-SSI
27 % Plotting untuk DD-SSI
28 - figure(1)
29 - axis([0,60,0,n])
```

2. truss12solve.m

Editor - D:\tesis\analisis Matlab\truss12solve.m

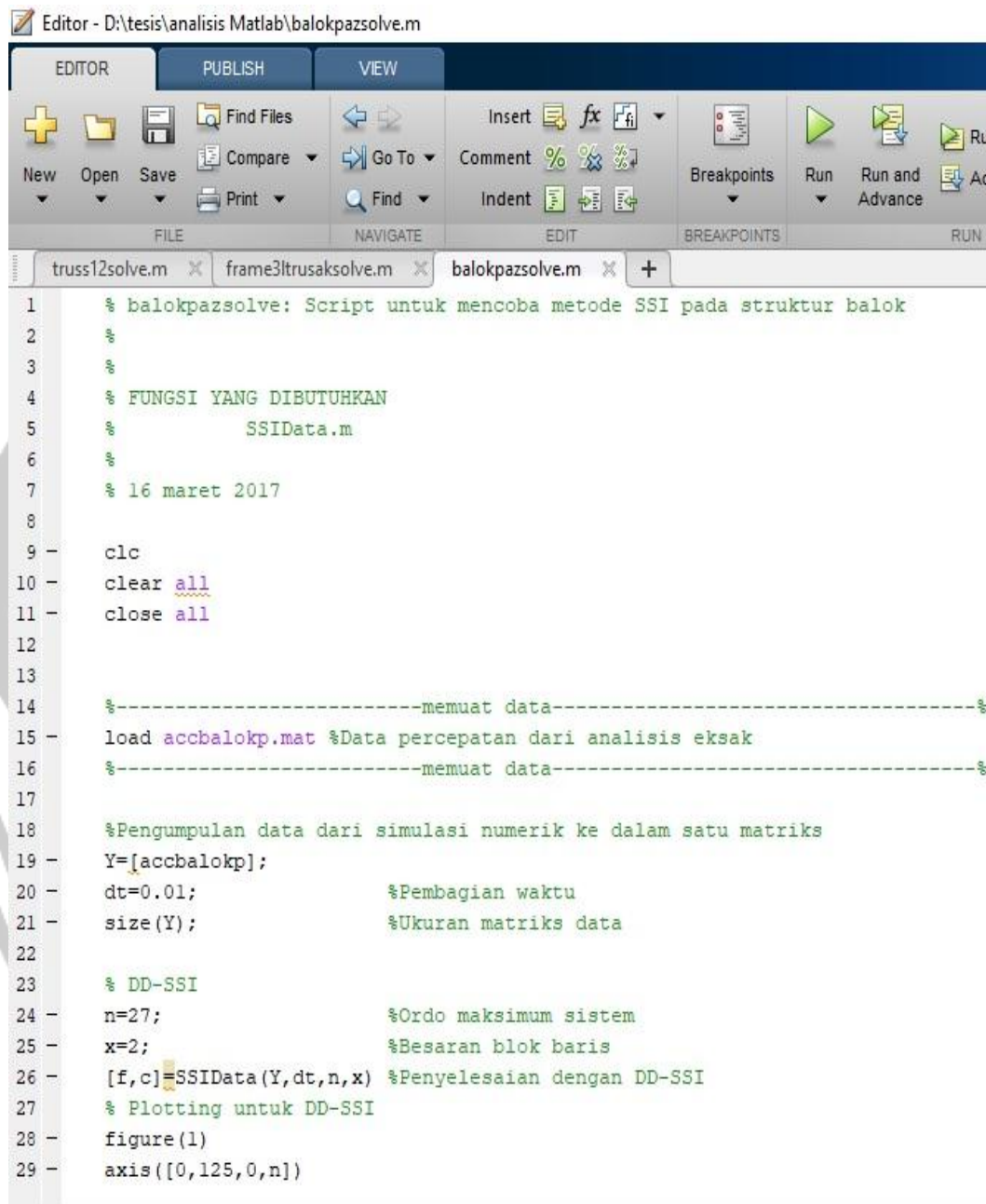


```

1  % truss12solve: Script untuk mencoba metode SSI pada struktur rangka batang
2  %
3  %
4  % FUNGSI YANG DIBUTUHKAN
5  %     SSIData.m
6  %
7  % 16 maret 2017
8
9  clc
10 clear all
11 close all
12
13
14 %-----memuat dan menyusun data-----
15 load acctruss12.mat %data percepatan dari analisis eksak
16
17 %Pengumpulan data dari simulasi numerik ke dalam satu matriks
18 Y=[acctruss12];
19 dt=0.01;           %Pembagian waktu
20 size(Y)           %Ukuran matriks data
21
22 % DD-SSI
23 n=60;             %Ordo maksimum sistem
24 x=4;             %Besaran blok baris
25 [f,c]=SSIData(Y,dt,n,x) %Penyelesaian dengan DD-SSI
26
27 % Plotting untuk DD-SSI
28 figure(1)
29 axis([0,900,0,n])

```

3. balokpazzsolve.m



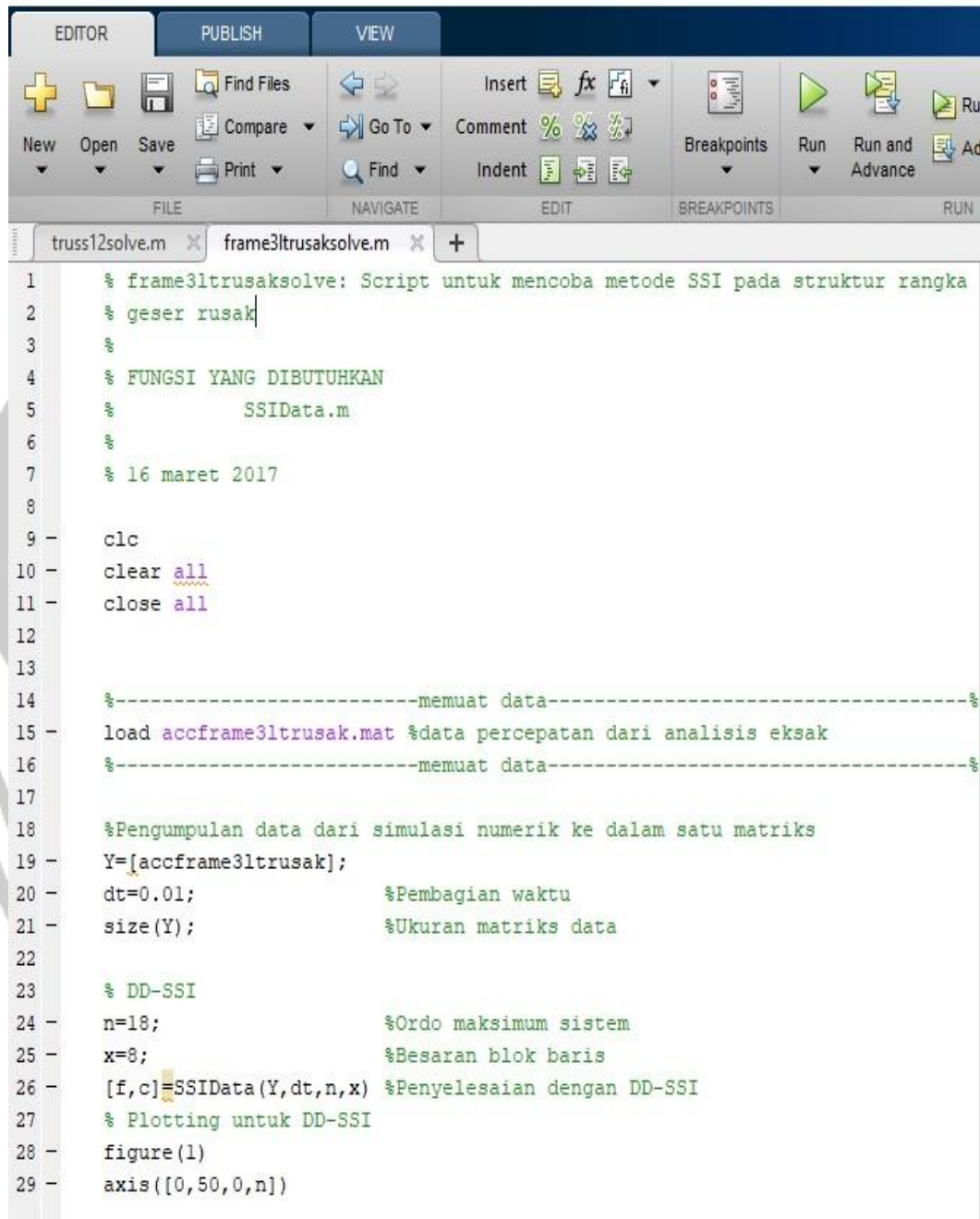
```
Editor - D:\tesis\analisis Matlab\balokpazzsolve.m

EDITOR PUBLISH VIEW
+ Find Files Insert fx fi
New Open Save Compare Go To Comment % % %
Print Find Indent Breakpoints Run Run and Advance Ad
FILE NAVIGATE EDIT BREAKPOINTS RUN

truss12solve.m x frame3ltrasaksolve.m x balokpazzsolve.m x +
1 % balokpazzsolve: Script untuk mencoba metode SSI pada struktur balok
2 %
3 %
4 % FUNGSI YANG DIBUTUHKAN
5 %     SSIData.m
6 %
7 % 16 maret 2017
8
9 - clc
10 - clear all
11 - close all
12
13
14 %-----memuat data-----%
15 - load accbalokp.mat %Data percepatan dari analisis eksak
16 %-----memuat data-----%
17
18 %Pengumpulan data dari simulasi numerik ke dalam satu matriks
19 - Y=[accbalokp];
20 - dt=0.01; %Pembagian waktu
21 - size(Y); %Ukuran matriks data
22
23 % DD-SSI
24 - n=27; %Ordo maksimum sistem
25 - x=2; %Besaran blok baris
26 - [f,c]=SSIData(Y,dt,n,x) %Penyelesaian dengan DD-SSI
27 % Plotting untuk DD-SSI
28 - figure(1)
29 - axis([0,125,0,n])
```

4. frame3ltrusaksolve.m

Editor - D:\tesis\analisis Matlab\frame3ltrusaksolve.m



```

1  % frame3ltrusaksolve: Script untuk mencoba metode SSI pada struktur rangka
2  % geser rusak
3  %
4  % FUNGSI YANG DIBUTUHKAN
5  %     SSIData.m
6  %
7  % 16 maret 2017
8
9  clc
10 clear all
11 close all
12
13
14 %-----memuat data-----%
15 load accframe3ltrusak.mat %data percepatan dari analisis eksak
16 %-----memuat data-----%
17
18 %Pengumpulan data dari simulasi numerik ke dalam satu matriks
19 Y=[accframe3ltrusak];
20 dt=0.01;           %Pembagian waktu
21 size(Y);           %Ukuran matriks data
22
23 % DD-SSI
24 n=18;               %Ordo maksimum sistem
25 x=8;                %Besaran blok baris
26 [f,c]=SSIData(Y,dt,n,x) %Penyelesaian dengan DD-SSI
27 % Plotting untuk DD-SSI
28 figure(1)
29 axis([0,50,0,n])

```

5. truss12rusaksolve.m

Editor - D:\tesis\analisis Matlab\truss12rusaksolve.m*

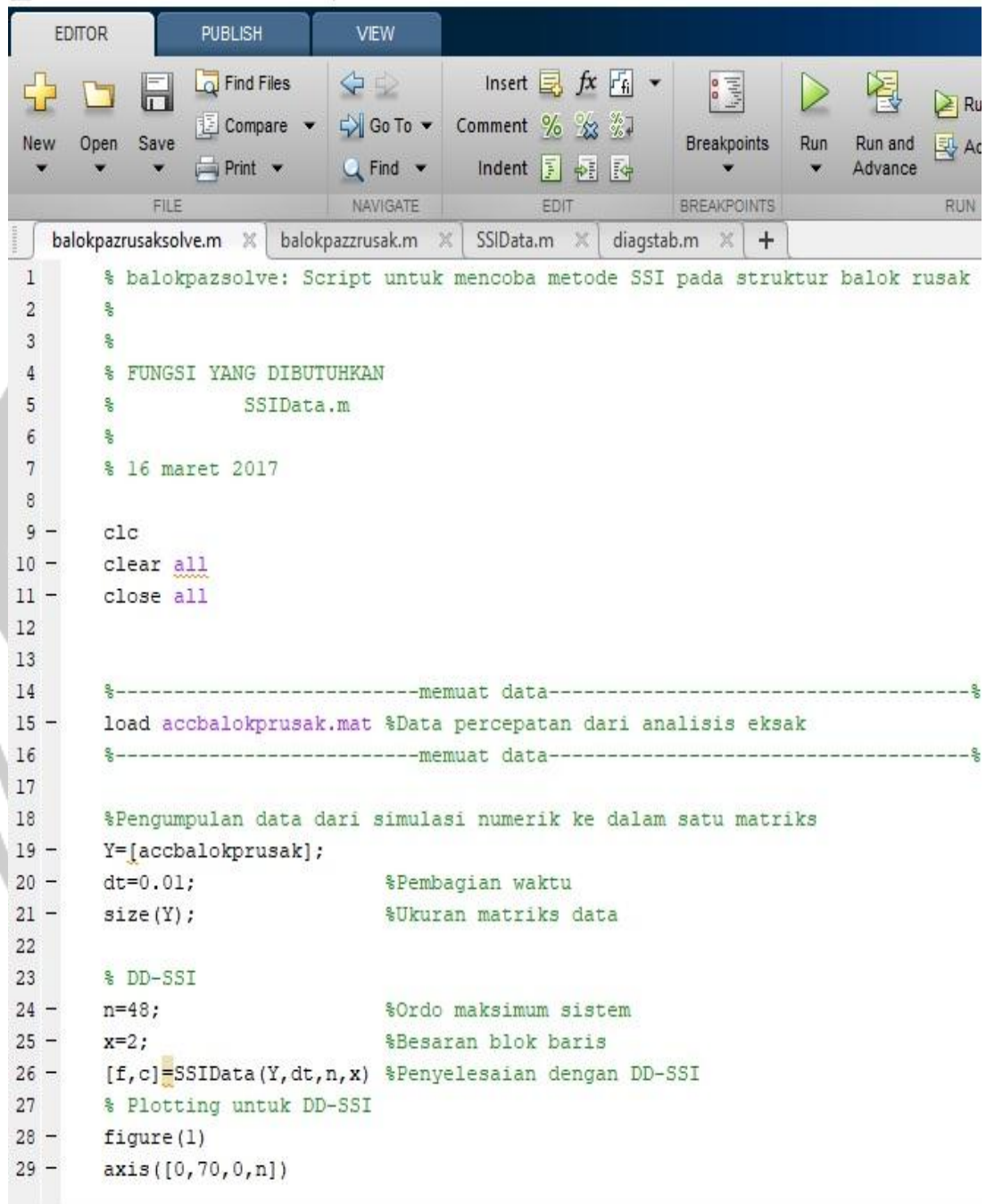
```

1  % truss12solve: Script untuk mencoba metode SSI pada struktur rangka batang
2  % rusak
3  %
4  % FUNGSI YANG DIBUTUHKAN
5  %     SSIData.m
6  %
7  % 16 maret 2017
8
9  clc
10 clear all
11 close all
12
13
14 %-----memuat dan menyusun data-----
15 load acctruss12rusak.mat %Data percepatan dari analisis eksak
16
17 %Pengumpulan data dari simulasi numerik ke dalam satu matriks
18 Y=[acctruss12rusak];
19 dt=0.01; %Pembagian waktu
20 size(Y) %Ukuran matriks data
21
22 % DD-SSI
23 n=60; %Ordo maksimum sistem
24 x=4; %Besaran blok baris
25 [f,c]=SSIData(Y,dt,n,x) %Penyelesaian dengan DD-SSI
26
27 % Plotting untuk DD-SSI
28 figure(1)
29 axis([0,500,0,n])

```


6. balokpazrusaksolve.m

Editor - D:\tesis\analisis Matlab\balokpazrusaksolve.m



```

1      % balokpazrusaksolve: Script untuk mencoba metode SSI pada struktur balok rusak
2      %
3      %
4      % FUNGSI YANG DIBUTUHKAN
5      %     SSIData.m
6      %
7      % 16 maret 2017
8
9      clc
10     clear all
11     close all
12
13
14     %-----memuat data-----%
15     load accbalokprusak.mat %Data percepatan dari analisis eksak
16     %-----memuat data-----%
17
18     %Pengumpulan data dari simulasi numerik ke dalam satu matriks
19     Y=[accbalokprusak];
20     dt=0.01;           %Pembagian waktu
21     size(Y);           %Ukuran matriks data
22
23     % DD-SSI
24     n=48;              %Ordo maksimum sistem
25     x=2;               %Besaran blok baris
26     [f,c]=SSIData(Y,dt,n,x) %Penyelesaian dengan DD-SSI
27     % Plotting untuk DD-SSI
28     figure(1)
29     axis([0,70,0,n])

```

1. output frame3lt.m

```
Command Window

Klat =

    1.0e+04 *
    5.2865   -2.7751    0.3998
   -2.7751    3.7778   -1.6015
    0.3998   -1.6015    1.2528

M =

    25     0     0
     0    25     0
     0     0    25

rd =

    0.0200

C =


    217.3189  -108.7198    15.6648
   -108.7198   158.2152   -62.7410
    15.6648   -62.7410    59.2913

>> wo

wo =

    10.2100
    30.8675
    55.4057
```

2. output truss12.m

 Command Window

wo =

```
96.3865
154.4246
217.8343
294.2669
381.7215
443.3073
607.8359
630.7979
654.5988
707.3499
843.1604
897.4890
```

worder =

```
10
11
12
9
8
7
6
5
4
3
2
1
```

rd =

0.0200

3. output balokpazz.m

Command Window

Kver =

1.0e+06 *

1.1540	-0.7344	0.3147
-0.7344	0.9180	-0.7344
0.3147	-0.7344	1.1540

Mver =

201.8634	38.5714	20.9938
38.5714	192.6786	38.5714
20.9938	38.5714	201.8634

rd =

0.0200

>> wo

wo =

21.5180
68.1196
120.7158

>> mode

mode =

1.0000	1.0000	1.0000
-1.3720	0.0000	1.8153
1.0000	-1.0000	1.0000

4. output frame3ltrusak.m

 Command Window

Klat =

36331	-13108	731.61
-13108	7762.1	-1377.1
731.61	-1377.1	821.69

M =

25	0	0
0	25	0
0	0	25

wo =

3.6925
11.146
40.727

worder =

3
2
1

rd =

0.02

5. output truss12rusak.m

```
Command Window

T1 =

    0.097781

rd =

    0.02

Warning: Simulation will start at a nonzero initial time.
> In ctrlMsgUtils.warning (line 25)
   In DynamicSystem.checkLsimInputs (line 90)
   In DynamicSystem/lsim (line 67)
   In truss12rusak (line 199)
Warning: The input signal is undersampled. Use a sampling period smaller
> In ctrlMsgUtils.warning (line 25)
   In ltipack.ssdata/lsim>LocalCheckSampling (line 157)
   In ltipack.ssdata/lsim (line 48)
   In ltipack.SystemArray/lsim_ (line 159)
   In DynamicSystem/lsim (line 77)
   In truss12rusak (line 199)
>> wo

wo =

    64.258
    102.95
    145.22
    196.18
    254.48
    295.54
    405.22
    420.53
     436.4
    471.57
    562.11
    598.33
```

6. output balokpazrusak.m

Command Window

Kver =

57701	-36719	15737
-36719	45898	-36719
15737	-36719	57701

Mver =

40.373	7.7143	4.1988
7.7143	38.536	7.7143
4.1988	7.7143	40.373

rd =

0.02

>> wo

wo =

10.759
34.06
60.358

>> mode

mode =

1	1	1
-1.372	1.8254e-16	1.8153
1	-1	1

7. output frame3ltsolve.m

```

Command Window

f =

Columns 1 through 13

    0.1063    10.1826    10.2402    10.1674    31.1666    55.6901    55.6999    55.7022    129.8872    133.5187    187.7658    211.9294    237.5523
    0    10.1826    10.2402    10.1674    31.1666    55.6901    55.6999    55.7022    129.8872    133.5187    187.7658    211.9294    237.5523
    0    0    7.2207    31.1858    10.1749    10.1689    10.1665    10.1652    55.6942    266.8069    133.2659    140.7405    143.0727
    0    0    0    31.1858    10.1749    10.1689    10.1665    10.1652    55.6942    55.6870    133.2659    140.7405    143.0727
    0    0    0    0    19.4396    30.8424    30.8432    30.8438    10.1646    55.6870    55.6832    10.1664    10.1648
    0    0    0    0    0    30.8424    30.8432    30.8438    10.1646    10.1692    55.6832    10.1664    10.1648
    0    0    0    0    0    0    65.4750    103.8606    30.8411    10.1692    10.1657    30.8428    30.8358
    0    0    0    0    0    0    0    103.8606    30.8411    30.8417    10.1657    30.8428    30.8358
    0    0    0    0    0    0    0    0    63.4289    30.8417    30.8407    55.6955    55.6629
    0    0    0    0    0    0    0    0    0    72.6154    30.8407    55.6955    55.6629
    0    0    0    0    0    0    0    0    0    72.3818    82.9509    85.8778
    0    0    0    0    0    0    0    0    0    0    82.9509    85.8778
    0    0    0    0    0    0    0    0    0    0    0    184.5128
    0    0    0    0    0    0    0    0    0    0    0    0    0
    0    0    0    0    0    0    0    0    0    0    0    0    0

Columns 14 through 15

140.4756    141.6005
140.4756    141.6005
192.0287    201.9149
192.0287    201.9149
272.4444    265.3061
272.4444    265.3061
10.1700    10.1740
10.1700    10.1740
55.6624    30.8439
55.6624    30.8439
30.8372    55.6537
30.8372    55.6537
67.0943    160.0137
67.0943    70.2123
0    70.2123

fc

```

```

Command Window

c =

Columns 1 through 13

    1.0000    0.0494    0.0451    0.0431    0.0694    0.1120    0.1120    0.1122    0.2476    0.1185    0.4711    0.3855    0.3029
    0    0.0494    0.0451    0.0431    0.0694    0.1120    0.1120    0.1122    0.2476    0.1185    0.4711    0.3855    0.3029
    0    0    1.0000    0.0739    0.0433    0.0431    0.0432    0.0435    0.1122    1.0000    0.1434    0.1271    0.1056
    0    0    0    0.0739    0.0433    0.0431    0.0432    0.0435    0.1122    0.1120    0.1434    0.1271    0.1056
    0    0    0    0    1.0000    0.0647    0.0648    0.0649    0.0435    0.1120    0.1121    0.0436    0.0437
    0    0    0    0    0    0.0647    0.0648    0.0649    0.0435    0.0430    0.1121    0.0436    0.0437
    0    0    0    0    0    0    1.0000    0.9989    0.0648    0.0430    0.0435    0.0651    0.0651
    0    0    0    0    0    0    0    0.9989    0.0648    0.0649    0.0435    0.0651    0.0651
    0    0    0    0    0    0    0    0    1.0000    0.0649    0.0650    0.1122    0.1124
    0    0    0    0    0    0    0    0    0    1.0000    0.0650    0.1122    0.1124
    0    0    0    0    0    0    0    0    0    0    1.0000    0.7167    0.6543
    0    0    0    0    0    0    0    0    0    0    0    0.7167    0.6543
    0    0    0    0    0    0    0    0    0    0    0    0    1.0000
    0    0    0    0    0    0    0    0    0    0    0    0    0
    0    0    0    0    0    0    0    0    0    0    0    0    0

Columns 14 through 15

    0.1099    0.0972
    0.1099    0.0972
    0.3554    0.3734
    0.3554    0.3734
    0.5063    0.4964
    0.5063    0.4964
    0.0429    0.0428
    0.0429    0.0428
    0.1123    0.0653
    0.1123    0.0653
    0.0652    0.1123
    0.0652    0.1123
    0.7357    1.0000
    0.7357    0.5721
    0    0.5721

fc

```


9. output balokpazzsolve.m

Frekuensi alami

Columns 25 through 27

288.02	288.05	288.06
288.02	288.05	288.06
247.31	247.31	247.32
247.31	247.31	247.32
201.99	201.99	201.98
201.99	201.99	201.98
21.487	147.23	147.25
21.487	147.23	147.25
11.64	169.67	169.65
11.64	169.67	169.65
67.078	21.459	120.46
67.078	21.459	120.46
92.906	11.728	92.909
92.906	11.728	92.909
147.24	120.48	105.05
147.24	120.48	105.05
120.45	92.901	67.021
120.45	92.901	67.021
169.69	67.022	21.444
169.69	67.022	21.444
328.9	321.68	11.732
101.21	329.04	11.732
101.21	3470.2	34.667
3347.1	203.17	319.85
409.49	87.436	329.03
0	87.436	3283.1
0	0	362.71

Rasio redaman

Columns 25 through 27

0.034899	0.034855	0.034822
0.034899	0.034855	0.034822
0.047337	0.047316	0.047312
0.047337	0.047316	0.047312
0.047475	0.047415	0.04728
0.047475	0.047415	0.04728
0.045422	0.088899	0.088909
0.045422	0.088899	0.088909
0.1497	0.20446	0.20481
0.1497	0.20446	0.20481
0.21739	0.045545	0.11692
0.21739	0.045545	0.11692
0.14299	0.14729	0.14258
0.14299	0.14729	0.14258
0.0891	0.11738	0.20489
0.0891	0.11738	0.20489
0.11764	0.14294	0.21699
0.11764	0.14294	0.21699
0.204	0.21704	0.046111
0.204	0.21704	0.046111
0.29598	0.21494	0.1483
0.56675	0.29737	0.1483
0.56675	1	1
0.99559	1	0.18773
0.64142	0.58313	0.29723
0	0.58313	1
0	0	1

10. output frame3ltrusaksolve.m

Frekuensi alami

Columns 13 through 18

320.75	320.41	315.71	295.1	295.54	291.23
431.49	444.54	345.74	295.1	295.54	291.23
118.65	115.87	237.07	259.14	285.95	254.95
3.6758	115.87	71.258	259.14	285.95	254.95
3.6758	71.189	71.258	71.47	72.655	74.947
11.114	71.189	3.6763	71.47	72.655	74.947
11.114	3.6759	3.6763	3.6759	3.674	3.6701
40.607	3.6759	11.116	3.6759	3.674	3.6701
40.607	11.115	11.116	11.115	11.115	11.114
71.805	11.115	40.643	11.115	11.115	11.114
71.805	40.632	40.643	55.368	40.696	40.686
63.359	40.632	57.33	55.368	40.696	40.686
63.359	59.027	57.33	40.668	55.156	56.563
0	59.027	101.61	40.668	55.156	56.563
0	0	101.61	68	73.797	91.605
0	0	0	68	73.797	91.605
0	0	0	0	62.826	63.693
0	0	0	0	0	63.693

Rasio redaman

Columns 13 through 18

0.20171	0.19659	0.099	0.14403	0.13226	0.10152
0.68549	0.70751	0.41755	0.14403	0.13226	0.10152
1	0.961	1	0.90666	0.82124	0.56459
0.034028	0.961	0.16579	0.90666	0.82124	0.56459
0.034028	0.16752	0.16579	0.15589	0.16242	0.12973
0.06898	0.16752	0.034062	0.15589	0.16242	0.12973
0.06898	0.034113	0.034062	0.034026	0.034663	0.035774
0.2206	0.034113	0.069008	0.034026	0.034663	0.035774
0.2206	0.069028	0.069008	0.069072	0.069022	0.068874
0.21008	0.069028	0.22057	0.069072	0.069022	0.068874
0.21008	0.22058	0.22057	0.33559	0.22024	0.22031
0.41996	0.22058	0.33269	0.33559	0.22024	0.22031
0.41996	0.38602	0.33269	0.22005	0.32278	0.34486
0	0.38602	0.99012	0.22005	0.32278	0.34486
0	0	0.99012	0.94395	0.86336	0.75589
0	0	0	0.94395	0.86336	0.75589
0	0	0	0	1	0.98651
0	0	0	0	0	0.98651

11. output truss12rusaksolve.m

Frekuensi alami

Command Window

Columns 49 through 60

64.028	63.915	64.034	64.227	64.239	64.194	64.133	64.182	64.196	64.3	64.231	64.239
64.028	63.915	64.034	64.227	64.239	64.194	64.133	64.182	64.196	64.3	64.231	64.239
102.78	102.74	102.78	43.048	102.82	102.72	102.72	102.8	102.88	102.8	102.93	102.81
102.78	102.74	102.78	43.048	102.82	102.72	102.72	102.8	102.88	102.8	102.93	102.81
90.525	90.521	90.533	102.83	90.544	90.544	90.554	90.552	90.556	43.034	90.554	90.575
90.525	90.521	90.533	102.83	90.544	90.544	90.554	90.552	90.556	43.034	90.554	90.575
145.47	145.6	145.73	90.542	145.6	145.64	145.67	43.01	43.048	90.561	145.51	145.63
145.47	145.6	145.73	90.542	145.6	145.64	145.67	43.01	43.048	90.561	145.51	145.63
134.8	134.81	134.8	37.882	134.83	134.83	134.81	145.5	145.55	145.57	134.79	134.82
134.8	134.81	134.8	37.882	134.83	134.83	134.81	145.5	145.55	145.57	134.79	134.82
171.32	171.31	171.29	27.127	171.3	171.28	171.26	134.81	134.82	134.83	171.24	171.26
171.32	171.31	171.29	27.127	171.3	171.28	171.26	134.81	134.82	134.83	171.24	171.26
196.35	196.64	196.61	145.63	196.55	196.6	196.61	171.29	171.29	171.28	296.94	196.74
196.35	196.64	196.61	145.63	196.55	196.6	196.61	171.29	171.29	171.28	296.94	196.74
296.89	296.88	296.89	134.82	261.24	296.92	296.92	196.7	296.91	196.72	196.9	297.02
296.89	296.88	296.89	134.82	261.24	296.92	296.92	196.7	296.91	196.72	196.9	297.02
261.23	261.19	261.3	68.282	296.86	261.31	261.2	296.93	196.57	207.7	261.31	261.32
261.23	261.19	261.3	68.282	296.86	261.31	261.2	296.93	196.57	207.7	261.31	261.32
294.28	290.31	291.78	110.05	250.96	250.06	295.16	261.28	261.32	296.87	295.01	291.94
294.28	290.31	291.78	110.05	250.96	250.06	295.16	261.28	261.32	296.87	295.01	291.94
248.3	250.37	249.54	171.31	289.72	290.41	249.24	250.49	290.2	261.12	249.51	248.98
248.3	250.37	249.54	171.31	289.72	290.41	249.24	250.49	290.2	261.12	249.51	248.98
207.67	207.75	207.54	154.78	208.09	235.28	207.44	290.79	250.28	289.34	207.55	207.17
207.67	207.75	207.54	154.78	208.09	235.28	207.44	290.79	250.28	289.34	207.55	207.17
233.84	317.59	235.19	196.68	238.22	207.47	234.5	207.92	207.54	249.91	236.37	233.73
233.84	317.59	235.19	196.68	238.22	207.47	234.5	207.92	207.54	249.91	236.37	233.73
267.94	236.21	221.24	190.42	231.32	224.57	277.31	237.55	233	193.35	225.05	220.25
267.94	236.21	221.24	190.42	231.32	224.57	277.31	237.55	233	193.35	225.05	220.25
223.93	229.01	186.97	207.76	316.68	190.56	224.21	230.23	238.22	316.39	189.63	187.55
223.93	229.01	186.97	207.76	316.68	190.56	224.21	230.23	238.22	316.39	189.63	187.55
192.22	192.76	43.036	296.94	43.049	317.59	190.79	190.8	189.02	233.63	279.9	43.006
192.22	192.76	43.036	296.94	43.049	317.59	190.79	190.8	189.02	233.63	279.9	43.006
43.022	43.047	151.95	261.29	199.01	43.056	43.007	38.012	218.47	227.71	43.075	251.41
43.022	43.047	151.95	261.29	199.01	43.056	43.007	38.012	218.47	227.71	43.075	251.41
162.26	156.5	151.95	261.29	37.841	156.74	154.98	156.29	156.2	246.31	156.16	299.24
162.26	156.5	151.95	261.29	37.841	156.74	154.98	156.29	156.2	246.31	156.16	299.24
162.26	37.809	250.68	249.7	158.46	249.54	154.98	156.29	156.2	38.171	156.16	299.24
38.078	37.809	320.82	291.78	158.46	249.54	38.103	27.543	38.01	38.171	243.08	153.31
38.078	108.52	37.875	291.78	26.711	38.018	38.103	217.97	38.01	164.49	243.08	153.31
27.26	108.52	37.875	235.04	45.504	38.018	111.65	217.97	27.599	164.49	37.93	38.071
117.01	27.21	107.14	235.04	69.132	113.37	111.65	112.47	120.8	27.489	37.93	38.071
117.01	71.294	107.14	228.35	69.132	113.37	27.167	112.47	120.8	138.7	27.577	27.466
72.709	71.294	27.222	228.35	111.89	27.056	225.84	69.552	71.849	138.7	116.56	103.75
72.709	49.961	65.304	78.698	111.89	68.334	225.84	69.552	71.849	40.642	116.56	103.75
59.936	210.2	65.304	237.91	194.32	68.334	67.706	57.458	52.821	68.484	41.632	41.654
86.672	210.2	49.807	237.91	194.32	58.49	67.706	57.458	68.392	68.484	66.857	63.975
222.89	85.884	328.39	320.12	78.19	58.49	55.656	318.97	68.392	102.46	66.857	63.975
222.89	135.68	74.046	130.86	144.78	73.02	55.656	321.25	107.31	102.46	106.51	174.85
3118.5	3172.2	364.15	329.59	144.78	332.82	76.532	326.55	107.31	323.78	106.51	174.85
343.26	389.05	3533.4	316.68	155.94	168.03	334.17	90.003	318.56	77.485	169.43	79.625
0	344.12	136.78	316.68	3270.1	168.03	351.11	126.78	319.85	77.485	169.43	79.625
0	0	136.78	2788.7	313.52	241.05	179.92	126.78	165.44	332.13	85.892	136.78
0	0	0	375.33	313.52	241.05	179.92	136.89	222.19	114.73	85.892	136.78
0	0	0	0	357.44	332.1	3062.9	242.2	222.19	169.74	126.81	114.55
0	0	0	0	0	390.92	229.7	242.2	3159.2	169.74	126.81	114.55
0	0	0	0	0	0	229.7	3166.1	339.58	173.98	321.56	311.87
0	0	0	0	0	0	0	368.37	340.9	3119.2	321.56	311.87
0	0	0	0	0	0	0	0	340.9	408.67	3137.5	211.59
0	0	0	0	0	0	0	0	0	370.43	416.07	3081.3
0	0	0	0	0	0	0	0	0	0	371.35	480.47
0	0	0	0	0	0	0	0	0	0	372.25	372.25

