BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan bahasan pada hasil analisis dan eksperimen tersebut, maka dapat ditarik beberapa kesimpulan sebagai berikut :

1. Nilai parameter modal struktur terestimasi struktur truss rangka bidang kondisi normal secara berturut-turut untuk frekuensi alami dan ragam getar adalah 68,6925 rad/s dan 40,7983 sedangkan pada saat kondisi rusak diperoleh 50,9254 rad/s dan 30,9473.
2. Indikasi kerusakan struktur pada penelitian ini hanya didasarkan pada rasio frekuensi alami (r) yang mengalami penurunan sebesar 34,8884 %, sedangkan untuk ragam getar yang dihitung dengan nilai *modal assurance criterion* (MAC) tidak digunakan karena hanya memiliki satu nilai ragam getar.
3. Metode *second order blind identification* (SOBI) cukup akurat untuk estimasi parameter modal struktur yaitu untuk nilai frekuensi alami tetapi hanya dapat memprediksi rasio redaman yang sangat kecil. Ragam getar pertama dapat diperoleh dalam sekali percobaan tetapi ragam getar kedua dan ketiga harus mengalami beberapa kali percobaan agar sesuai dengan nilai hasil *finite element* (FE). Hal tersebut dikarenakan adanya *time lag* antar respons struktur yang dihasilkan sehingga menjadi kurang efektif . Metode ini juga memiliki kekurangan yaitu tidak ada tolok ukur ataupun

kontrol hasil parameter modal struktur terestimasi, serta hasil parameter modal terestimasi sangat sensitif terhadap nilai nfft yang digunakan.

5.2. Saran

Berdasarkan hasil kesimpulan dan pembahasan pada estimasi parameter modal model eksperimental maka penulis menyarankan beberapa poin penting yaitu :

1. Metode *second order blind identification* (SOBI) sudah cukup akurat tetapi diperlukan metode pembanding yang lebih stabil seperti *stochastic subspace identification* (SSI).
2. Diperlukan lebih dari satu sensor untuk merekam respons struktur guna menambah ragam getar prediksi dan menambah parameter indikator kerusakan struktur dengan nilai *modal assurance criterion* (MAC).
3. Jika menggunakan sensor lebih dari satu perhatikan juga sinkronisasi waktu antar sensor. Sinkronisasi waktu sangat penting untuk dilakukan karena akan berpengaruh pada estimasi ragam getar.
4. Pastikan sensor menggunakan batrai baru dan *memory card* sesuai standar produksi pabrik guna meminimalisir kesalahan pada saat perekaman data.

DAFTAR PUSTAKA

Arfiadi Y., 1996, *Pengembangan Program Bantu untuk Analisis Struktur dengan Menggunakan Matlab*, Laporan Penelitian, Program Studi Teknik Sipil Fakultas Teknik, Universitas Atma Jaya Yogyakarta, Yogyakarta.

Arfiadi Y., 2016a, *Analisis Struktur dengan Program Matlab dan FreeMat*, Cahaya Atma Pustaka, Kelompok Penerbit Universitas Atma Jaya Yogyakarta, Yogyakarta.

Arfiadi Y., 2016b, *Bahan Kuliah Dinamika Struktur Lanjut*, Universitas Atma Jaya Yogyakarta, Yogyakarta.

Belouchrani, A. dan Abed-meraim, K, 1997, Using Second-Order Statistics , *Journal IEEE Transactions on Signal Processing* , Vol 45, pp. 434–444.

Bernagozzi , G., Landi, L., dan Diotallevi, P.P., 2016, On the Output-Only Vibration-Based Damage Detection of Frame Structures, *Proceeding of Structural Health Monitoring, Damage Detection & Mechatronics*, Vol 7, pp. 23-33

Brewick, P. T. dan Smyth, A. W, 2014, On the Application of Blind Source Separation for Damping Estimation of Bridges Under Traffic Loading, *Journal of Sound and Vibration*, Vol 333, pp. 7333–7351.

Brincker, R. dan Ventura, C, 2015, *Introduction to Oprational Modal Analysis*, John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, West Sussex, PO19 8SQ, United Kingdom.

Chen, H.L, Spyrakos, C.C., dan Venkatesh, G., Evaluating Structural Deterioration by Dynamic Responsse, Journal of Structural Engineering, 1995. 121(8): p. 1197-1204.

Frans, R. dan Arfiadi, Y, 2017. Sistem Identifikasi Struktur dengan Menggunakan Metode *Frequency Domain Decomposition-Natural Excitation Technique*, KonTeks 11, Universitas Tarumanegara.

Graces, F.,2008, Identification of Civil Engineering Structures, *Engineering Sciences* [*physics*], Universite Paris-Est. <NNT : 2008PEST0238>. <tel-00470540>

Hetland, R., 2015, Operational Modal Analysis of the Dolmsundet Bridge During Construction, thesis of NTNU.

MATLAB and Statistics Toolbox Release 2015b, The MathWorks, Inc.,Natick, Massachusetts, United States.

Paz, M .,1986, *Microcomputer-Aid Engineering : Structural Dynamics*, Van Nostrand Reinhold, New York.

Paz, M .,1985, *Structural Dynamics: Theory and Computation*, Van Nostrand Reinhold, New York.

Pedoman Konstruksi dan Bangunan, 2009, *Pemeriksaan Jembatan Rangka Baja* (No. 005/BM/2009), Kementrian Pekerjaan Umum Direktorat Jendral Bina Marga, Jakarta.

Poncelet, F., Kerschen, G., Golinval, J.C., dan Verhelst, D.,2007, Output-only Modal Analysis using Blind Source Separation Techniques, *Journal Mechanical Systems and Signal Processing*, Vol 21, pp. 2335–2358.

Poncelet, F., 2010, Experimental Modal Analysis using Blind Source Separation Techniques, Ph.D Disertation, University of liege.

Rainieri, C. dan Fabbrocino, G., 2014, *Operational Modal Analysis of Civil Engineering Structure*. Springer, New York.

Sadhu, A., Hazra, B., dan Narasimhan, S., 2012, Blind Identification of Earthquake-Excited Structures, *Journal Smart Materials and Structures* , Vol 21, pp. 1-12.

S. Beskhyroun dan Q. Ma.,2012, Low-Cost Accelerometers for Experimental Modal Analysis, *15th World Conference on Earthquake Engineering*, Auckland, New Zealand.

S.S. Rao, 2004, *Mechanical Vibrations*, Pearson, Upper Saddle River, NJ, New York.

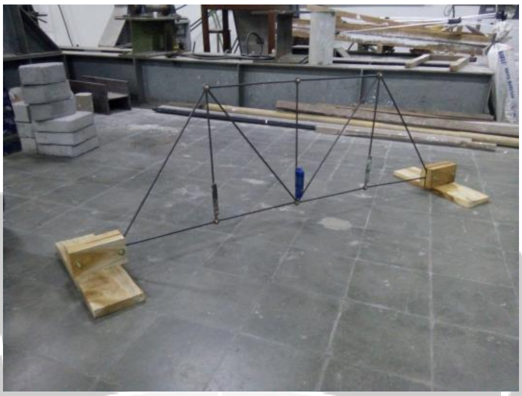
Schanke, SA., 2015, Operational Modal Analysis of Large Bridges, thesis of NTNU.

Spiridonakos, M.,Yadav. N., dan Chatzi, E.,2014, Identification and Damage Detection of Shear Frame Model Based on a Blind Source Separation Method, *7th European Workshop on Structural Health Monitoring*, July 8-11, Nantes, France.

Zhou, W. dan Chelidze, D., 2007, Blind Source Separation Based Vibration Mode Identification, *Journal Mechanical Systems and Signal Processing*, vol 21, pp. 3072–3087

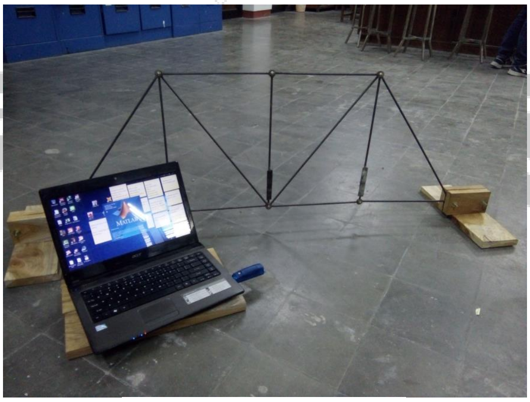
LAMPIRAN

PROSEDUR EKSPERIMEN



Struktur truss yang dikenai getaran *ambient*

Sensor yang memuat data rekaman respons struktur



Mengekstrak data

LAMPIRAN

ALAT DAN BAHAN EKSPERIMEN

Tulangan baja ϕ 6 mm

Kayu dukungan struktur

Sensor

Palu

Laptop

Sepana hujung terbuka

Tang kombinasi (plier)

Selotip

LAMPIRAN
INPUT DAN OUTPUT
STRUKTUR TRUSS

INPUT
(trussexp.m)

```
%---------Simulasi struktur truss rangka bidang------------------%
n1=coor(0,0)
n2=coor(0.3,0)
n3=coor(0.6,0)
n4=coor(0.9,0)
n5=coor(1.2,0)
n6=coor(0.3,0.4)
n7=coor(0.6,0.4)
n8=coor(0.9,0.4)

E= 2e8   % kN/m^2
D=0.006 % m
A1= 0.25*pi*D^2 %m2
mbar= 0.2826 %kg/m' %7850 kg/m3*A

[L1,T1]=memt(n1,n2); %--menghitung L dan T
k1=klt(E,A1,L1);  %--k lokal
K1=kg(k1,T1)  ;  %--K global
m1=[mbar*L1/3 0 mbar*L1/6 0  %--m lokal
    0 mbar*L1/3 0 mbar*L1/6
    mbar*L1/6 0 mbar*L1/3 0
    0 mbar*L1/6 0 mbar*L1/3];
M1=kg(m1,T1) %--M global
ID1=[0 0 1 2]; %-- vektor tujuan

A2=A1

[L2,T2]=memt(n2,n3);
k2=klt(E,A2,L2);
K2=kg(k2,T2);
m2=[mbar*L2/3 0 mbar*L2/6 0  %--m lokal
    0 mbar*L2/3 0 mbar*L2/6
    mbar*L2/6 0 mbar*L2/3 0
    0 mbar*L2/6 0 mbar*L2/3]
M2=kg(m2,T2) %--M global
ID2=[1 2 3 4];

A3=A1

[L3,T3]=memt(n3,n4);
k3=klt(E,A3,L3);
K3=kg(k3,T3);
m3=[mbar*L3/3 0 mbar*L3/6 0  %--m lokal
    0 mbar*L3/3 0 mbar*L3/6
    mbar*L3/6 0 mbar*L3/3 0
    0 mbar*L3/6 0 mbar*L3/3]
M3=kg(m3,T3) %--M global
ID3=[3 4 5 6];

A4=A1
```

```
[L4,T4]=memt(n4,n5);
k4=klt(E,A4,L4);
K4=kg(k4,T4);
m4=[mbar*L4/3 0 mbar*L4/6 0  %--m lokal
    0 mbar*L4/3 0 mbar*L4/6
    mbar*L4/6 0 mbar*L4/3 0
    0 mbar*L4/6 0 mbar*L4/3]
M4=kg(m4,T4) %--M global
ID4=[5 6 0 0];

A5=A1

[L5,T5]=memt(n6,n7);
k5=klt(E,A5,L5);
K5=kg(k5,T5);
m5=[mbar*L5/3 0 mbar*L5/6 0  %--m lokal
    0 mbar*L5/3 0 mbar*L5/6
    mbar*L5/6 0 mbar*L5/3 0
    0 mbar*L5/6 0 mbar*L5/3]
M5=kg(m5,T5) %--M global
ID5=[7 8 9 10];

A6=A1

[L6,T6]=memt(n7,n8);
k6=klt(E,A6,L6);
K6=kg(k6,T6);
m6=[mbar*L6/3 0 mbar*L6/6 0  %--m lokal
    0 mbar*L6/3 0 mbar*L6/6
    mbar*L6/6 0 mbar*L6/3 0
    0 mbar*L6/6 0 mbar*L6/3]
M6=kg(m6,T6) %--M global
ID6=[9 10 11 12];

A7=A1

[L7,T7]=memt(n1,n6);
k7=klt(E,A7,L7);
K7=kg(k7,T7);
m7=[mbar*L7/3 0 mbar*L7/6 0  %--m lokal
    0 mbar*L7/3 0 mbar*L7/6
    mbar*L7/6 0 mbar*L7/3 0
    0 mbar*L7/6 0 mbar*7/3]
M7=kg(m7,T7) %--M global
ID7=[0 0 7 8];

A8=A1

[L8,T8]=memt(n2,n6);
k8=klt(E,A8,L8);
K8=kg(k8,T8);
m8=[mbar*L8/3 0 mbar*L8/6 0  %--m lokal
    0 mbar*L8/3 0 mbar*L8/6
    mbar*L8/6 0 mbar*L8/3 0
    0 mbar*L8/6 0 mbar*8/3]
```

```
M8=kg(m8,T8) %--M global
ID8=[1 2 7 8];

A9=A1

[L9,T9]=memt(n3,n6);
k9=klt(E,A9,L9);
K9=kg(k9,T9);
m9=[mbar*L9/3 0 mbar*L9/6 0  %--m lokal
    0 mbar*L9/3 0 mbar*L9/6
    mbar*L9/6 0 mbar*L9/3 0
    0 mbar*L9/6 0 mbar*9/3]
M9=kg(m9,T9) %--M global
ID9=[3 4 7 8];

A10=A1

[L10,T10]=memt(n3,n7);
k10=klt(E,A10,L10);
K10=kg(k10,T10);
m10=[mbar*L10/3 0 mbar*L10/6 0  %--m lokal
    0 mbar*L10/3 0 mbar*L10/6
    mbar*L10/6 0 mbar*L10/3 0
    0 mbar*L10/6 0 mbar*10/3]
M10=kg(m10,T10) %--M global
ID10=[3 4 9 10];

A11=A1

[L11,T11]=memt(n3,n8);
k11=klt(E,A11,L11);
K11=kg(k11,T11);
m11=[mbar*L11/3 0 mbar*L11/6 0  %--m lokal
    0 mbar*L11/3 0 mbar*L11/6
    mbar*L11/6 0 mbar*L11/3 0
    0 mbar*L11/6 0 mbar*11/3]
M11=kg(m11,T11) %--M global
ID11=[3 4 11 12];

A12=A1

[L12,T12]=memt(n4,n8);
k12=klt(E,A12,L12);
K12=kg(k12,T12);
m12=[mbar*L12/3 0 mbar*L12/6 0  %--m lokal
    0 mbar*L12/3 0 mbar*L12/6
    mbar*L12/6 0 mbar*L12/3 0
    0 mbar*L12/6 0 mbar*12/3]
M12=kg(m12,T12) %--M global
ID12=[5 6 11 12];

A13=A1

[L13,T13]=memt(n5,n8);
```

```
k13=klt(E,A13,L13);
K13=kg(k13,T13);
m13=[mbar*L13/3 0 mbar*L13/6 0  %--m lokal
    0 mbar*L13/3 0 mbar*L13/6
    mbar*L13/6 0 mbar*L13/3 0
    0 mbar*L13/6 0 mbar*13/3]
M13=kg(m13,T13) %--M global
ID13=[0 0 11 12];

dof=12

K=asst(K1,ID1,dof);
K=K+asst(K2,ID2,dof);
K=K+asst(K3,ID3,dof);
K=K+asst(K4,ID4,dof);
K=K+asst(K5,ID5,dof);
K=K+asst(K6,ID6,dof);
K=K+asst(K7,ID7,dof);
K=K+asst(K8,ID8,dof);
K=K+asst(K9,ID9,dof);
K=K+asst(K10,ID10,dof);
K=K+asst(K11,ID11,dof);
K=K+asst(K12,ID12,dof);
K=K+asst(K13,ID13,dof);%-- K struktur

M=asst(M1,ID1,dof);
M=M+asst(M2,ID2,dof);
M=M+asst(M3,ID3,dof);
M=M+asst(M4,ID4,dof);
M=M+asst(M5,ID5,dof);
M=M+asst(M6,ID6,dof);
M=M+asst(M7,ID7,dof);
M=M+asst(M8,ID8,dof);
M=M+asst(M9,ID9,dof);
M=M+asst(M10,ID10,dof);
M=M+asst(M11,ID11,dof);
M=M+asst(M12,ID12,dof);
M=M+asst(M13,ID13,dof);%-- M struktur
nc=9
nv=3
Klat=kcon(K,nc,nv);
Mlat=kcon(M,nc,nv);
n=size(Klat)
n=n(1)  %--ukuran DOF atau n=2
N=2*n   %--ukuran state vector

eo=[-diag(Mlat)]

[eigv,eigval]=eig(Mlat\Klat);
[wo,worder]=sort(sqrt(diag(eigval)));
modeshape=eigv(:,worder)
for i=1:3
modes(:,i)=modeshape(:,i)/modeshape(3,i)
end
```

```
T1=2*pi/wo(1)   %---waktu getar

rd=0.02 %--rasio redaman 2%
am=rd*4*pi/T1 %---sebanding massa
C=am*Mlat  %---matriks redaman
%-------------------------State space Eq-------------------------%
A=[zeros(n,n) eye(n);-inv(Mlat)*Klat -inv(Mlat)*C];
E=[zeros(n,1);inv(Mlat)*eo];

%-------------------------Y=X-------------------------------------%
Cy=eye(N);
Dy=zeros(N,1);

syst1=ss(A,E,Cy,Dy);

t1=0:0.01:1000;  %time series
iu1=randn(1,length(t1))';

[y1,t1,z1]=lsim(syst1,iu1,t1);    %---simulasi
plot(t1,y1(:,1),'-k')

xlabel('waktu (detik)')
ylabel('Perpindahan (m)')

%----------------------perpindahan max----------------------------%

y1max=(max(abs(y1(:,1))))
y2max=(max(abs(y1(:,2))))
y3max=(max(abs(y1(:,3))))

%-----------------------------------------------------------------%
I0=[1;1;1]
acctrussexp=-[inv(Mlat)*Klat inv(Mlat)*C]*z1'-I0*iu1';

percepatan_struktur=acctrussexp(1,:);
save acctrussexp.mat
%-----------------------------------------------------------------%
```

INPUT
(inputtrussexp.m)

```
%--------------------Load data and arrange-----------------------%

%load acctrussexp.mat

load acctrussexp.mat

%Gather data from all measurement channels into one matrix
Y=[acctrussexp];
%Y=detrend(Y); %Remove linear trend
dt=0.01; %Time step
size(Y)
%-----------------------------------------------------------------%
```

INPUT
(solvesobi.m)

```
%---------------find frequency-damp ratio-modeshape-------------%

Required function :
Sobi.m
JAD.m
Sobifind.m

[A,S]=sobi(Y);
[f,c]=sobifind(S,dt);

%-----------------------------------------------------------------%
```

INPUT
(sobi.m)

```
%-------------------------------SOBI-------------------------------%

function [A,S]=sobi(Y)

%INPUT  Y:data metrix from time series
%OUTPUT A:Mixing matrix
%       S:Source
%Required function : JAD.m

%----------------------Langkah awal--------------------------------%
[l,N]=size(Y);  %Measurment channels and number of measurment
p=min(100,ceil(N/3)); %number of time lag
Y=Y-kron(mean(Y')',ones(1,N)); %make data zero mean

%----------------------whitening the data--------------------------%
[~,S,V]=svd(Y',0);  %SVD of the observed data
W=pinv(S)*V';       %Whitening matrix
Z=W*Y;              %Whitened data
clear S V Y
%--------------------correlation matrix----------------------------%
k=1;
for j=1:l:p*l
    k=k+1;
    Rxp=Z(:,k:N)*Z(:,1:N-k+1)'/(N-k+1)/1;
    Rz(:,j:j+l-1)=sqrt(sum(diag(Rxp'*Rxp)))*Rxp;
end;

%-------------joint approximate diagonalization--------------------%
[UA]=JAD(l,N,p,Rz);

%--------------mixing matrix and sources---------------------------%
A=pinv(W)*UA;   %mixing matrix
S=UA'*Z;  %sources
%------------------------------------------------------------------%

end
```

INPUT
(JAD.m)

```
function [UA]=JAD(l,N,p,Rz)
%JAD: perform joint approximation diagonalization

%INPUT  l: measurmen channels
%       N: number of measurments
%       p: number of time lags
%       Rz: correlation matrix

%output UA: unitary matrix
%------------------------initial step-----------------------------%
UA=eye(l);   %define for later user
t=1/sqrt(N); %treshold

check=1; %set to start
while check   %while under treshold
    check=0;
    for j=1:l-1,
        for k=j+1:l,
            j1=j:l:p*l;
            k1=k:l:p*l;
            G=[Rz(j,j1)-Rz(k,k1);Rz(j,k1)+Rz(k,j1);1i*(Rz(k,j1)-
Rz(j,k1))];

            [E,D]=eig(real(G*G'));
            [~,K]=sort(diag(D));

            angle=E(:,K(3));
            if angle(1)<0
                angle=-angle;
            end
            c=sqrt(0.5+angle(1)/2);
            s=0.5*(angle(2)-1i*angle(3))/c;
            if abs(s)>t
                check=1;
                index=[j;k];
                R=[c -conj(s);s c]; %rotation matrix

                %update the corelations matrix
                Rz(index,:)=R'*Rz(index,:);
                Rz(:,[j1 k1])=[c*Rz(:,j1)+s*Rz(:,k1), ...
                    -conj(s)*Rz(:,j1)+c*Rz(:,k1)];

                %update unitary matrix
                UA(:,index)=UA(:,index)*R;
            end
        end
    end
end
%-----------------------------------------------------------------%
end
```

INPUT
(sobifind.m)

```
function[f,c]=sobifind(S,dt)

%sobifind : find frequencies and damping ratio from source

% INPUT : Y=source from sobi
%         dt=time step
% OUTPUT : f=natural frequencies
%          c= damping ratio

%-----------------------initial step------------------------------%
[l,N]=size(S); %measurment channels and number of measurment
%nfft=49160; % number of fast fourir transform for experiment
nfft=2048*512; %number of fast fourir transform for FE
np=9; %number of peaks
freq=2*pi*[0:nfft-1].'*1/(dt*nfft); %rad/s

%---calculation of modal parameters for each measurment channel--%
for k=1:l
    g(:,k)=fft(S(k,:),nfft)/N;  %fast fourier transform
    [~,top]=max(abs(g(:,k)));  %find peak
    ns=floor(np/2); %number of surrounding value
    if top<(ns+1)  %to avoid negative ie
        ie=(1:np);
    else
        ie=(top-ns:top+ns);
    end

    A=[g(ie,k),ones(np,1)];
    B=1i*freq(ie).*g(ie,k);
    x=A\B;%solve system of linear equations

    lamda=x(1);%Eignvalue

    f(k,1)=abs(lamda) %natural freq (rad/s)

    c(k,1)=-real(lamda)./abs(lamda) %damping ratio

end
%-----------------------------------------------------------------%
end
```

INPUT
(ExpNor15mts.m)

```
%Load data of damage structure in experiment
load health15mts.mat %Load data

%Gather data from all measurement channels into one matrix
Y=[health15mts];
Y=Y';
Y=detrend(Y); %Remove linear trend
dt=0.02; %Time step
size(Y)
%------------------------------------------------------------------%
```

OUTPUT
(ExpNor15mts.m)

```
>> ExpNor15mts

ans =

           1       49326

%------------------------------------------------------------------%
```

INPUT
(solvesobi.m)

```
%---------------find frequency-damp ratio-modeshape--------------%

Required function :
Sobi.m
JAD.m
Sobifind.m %-----nfft==49160

[A,S]=sobi(Y);
[f,c]=sobifind(S,dt);
%------------------------------------------------------------------%
```

OUTPUT
(solvesobi.m)

```
>> solvesobi

f =

   68.6925

c =

  -1.1233e-05

>> A

A =

   40.7983
%------------------------------------------------------------------%
```

INPUT
(ExpDmg15mts.m)

```
%Load data of normal structure in experiment

load damage15mts.mat %Load data

%Gather data from all measurement channels into one matrix
Y=[damage15mts];
Y=Y';
Y=detrend(Y); %Remove linear trend
dt=0.02; %Time step
size(Y)
```

OUTPUT
(ExpDmg15mts.m)

```
>> ExpDmg15mts

ans =

        1       50676
```

INPUT
(solvesobi.m)

```
%---------------find frequency-damp ratio-modeshape--------------%

Required function :
Sobi.m
JAD.m
Sobifind.m %-----nfft==49160

[A,S]=sobi(Y);
[f,c]=sobifind(S,dt);
%-----------------------------------------------------------------%
```

OUTPUT
(solvesobi.m)

```
>> solvesobi

f =

      50.9254

c =

   1.0103e-04
>> A

A =

   33.9473
```

LOAD DATA

```
%--------------------------------------------------------------%
```

1. health15mts.mat
2. damage15mts.mat