Matlab ile Diferansiyel Denklem COZUMLERI

- Bir ya da daha fazla fonksiyonun türevlerini içeren denklemlere diferansiyel denklem denir.

- Bir yada daha fazla bağımlı değişken ve bir bağımsız değişkenden oluşan diferansiyel denklemlere adi diferansiyel denklem denir.

- Bir veya daha fazla bağımlı ve birden fazla bağımsız değişkenin olduğu diferansiyel denklemlere de kısmi diferansiyel denklem denir.

MATLAB ortamında diferansiyel denklemler hem sayısal hem de sembolik (analitik) çözülebilir

Sembolik Diferansiyel Denklem Terimleri

y	y
$\frac{dy}{dt}$	Dy
$\frac{d^2y}{dt^2}$	D2y
$\frac{d^ny}{dt^n}$	Dny

Verilen bagimsiz degisken *t* ve bagimli degisken $y(t)$,

Burada A_0, A_1, …, A_n, sabitlerdir.

$$A_n \frac{d^n y}{dt^n} + ... + A_1 \frac{dy}{dt} + A_0 y(t) = f(t)$$

Matlab’ta diferansiyel denklemlerin sembolik çözümleri “dsolve” komutu ile bulunur.

$$b_2 \frac{d^2 y}{dt^2} + b_1 \frac{dy}{dt} + b_0 y = A \sin at$$

$$y(0) = C_1 \qquad \text{and } y'(0) = C_2$$

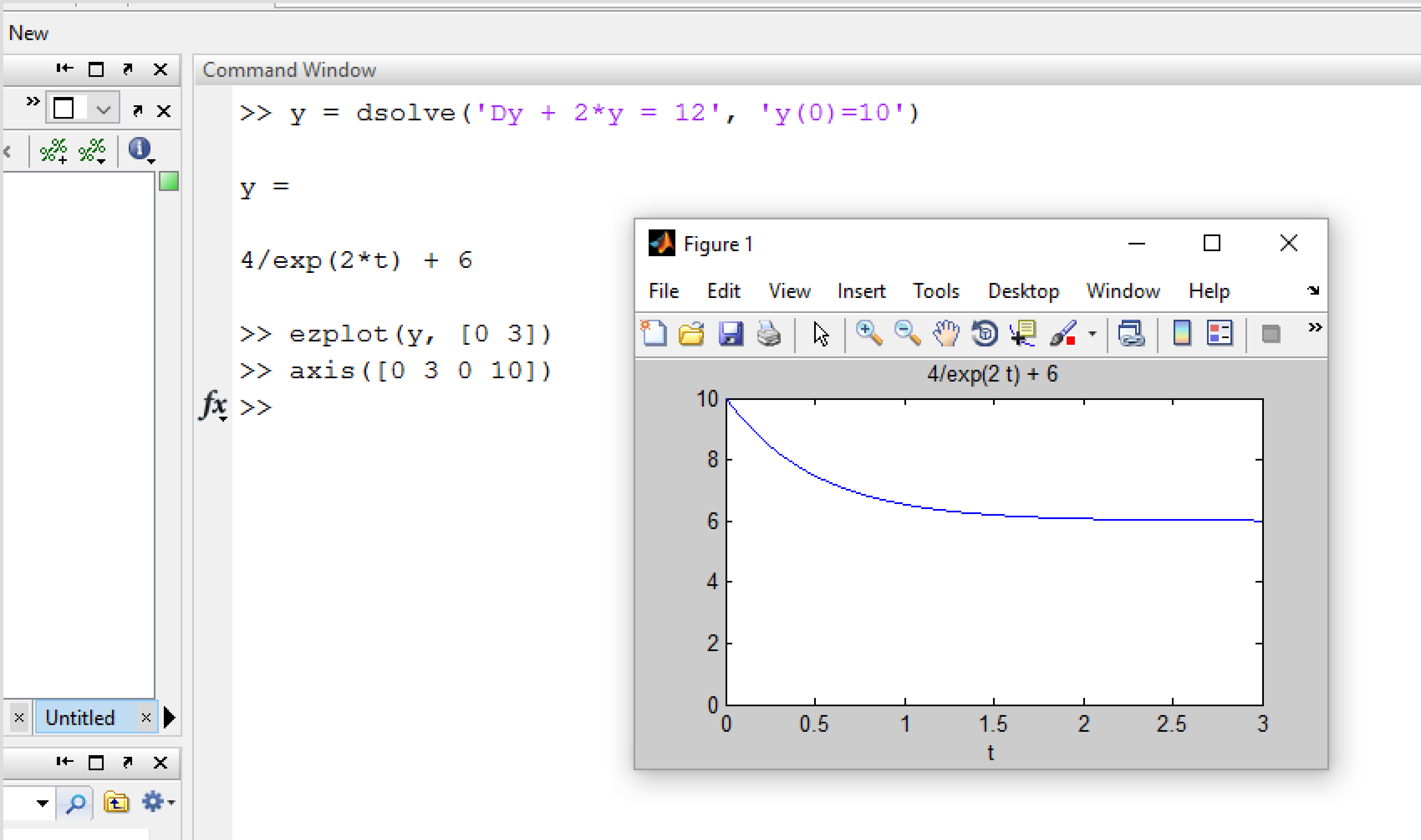
```
>> y = dsolve(‘b2*D2y+b1*D1y+b0*y=A*sin(a*t)’,
          ‘y(0)=C1’, ‘Dy(0)=C2’)

>> ezplot(y, [t1 t2])
```

$$x\frac{dy}{dx} + y = e^{2x} - \sin(x)$$

Command Window

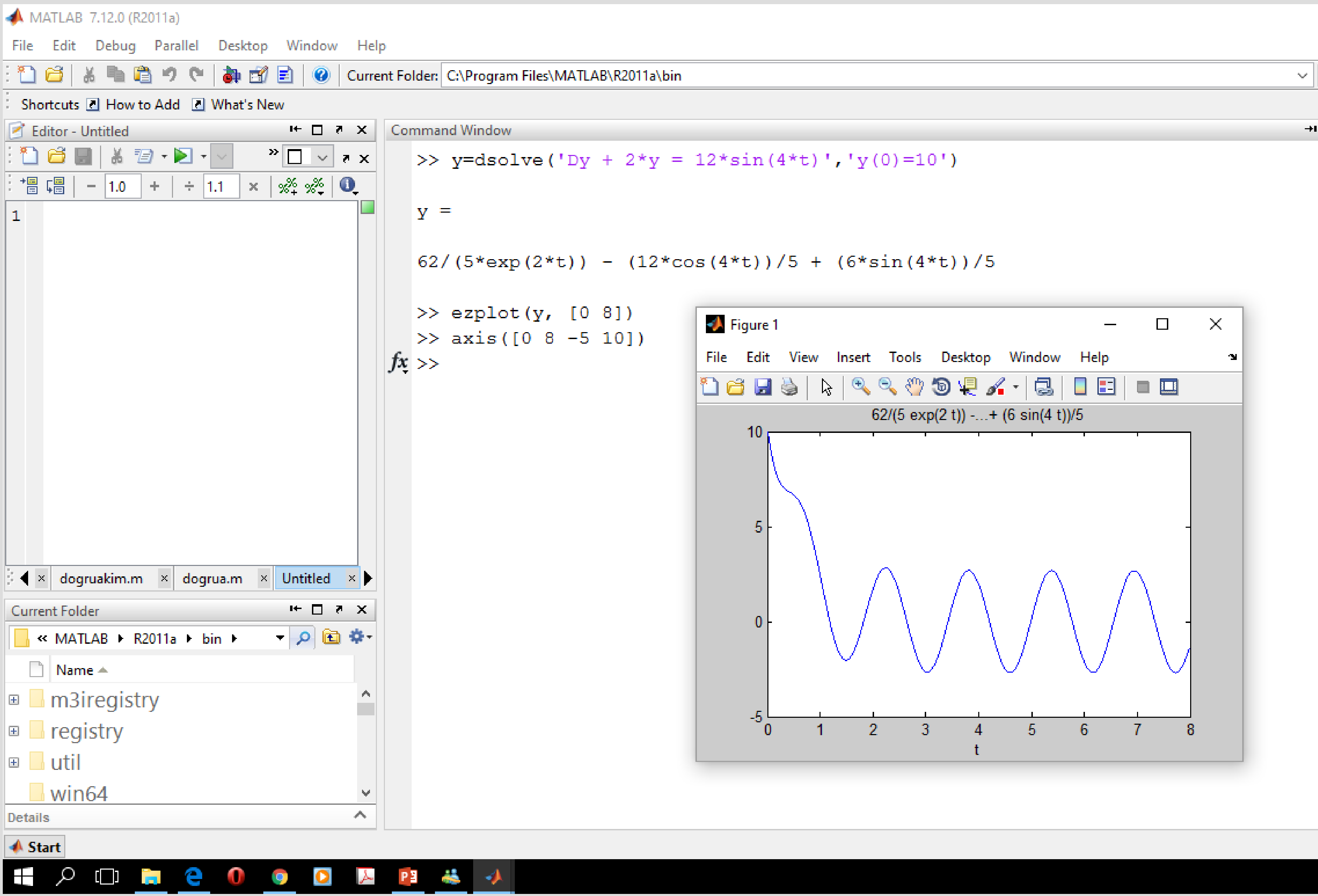
```
>> f=dsolve('x*Dy+y=exp(2*x)-sin(x)')

f =

exp(2*x) - sin(x) - C5/exp(t/x)

>> pretty(f)

                                 C5
  exp(2 x) - sin(x) - --------
                                / t \
                           exp| - |
                                \ x /
>> 
```

$$\frac{dy}{dt} + 2y = 12 \qquad y(0) = 10$$

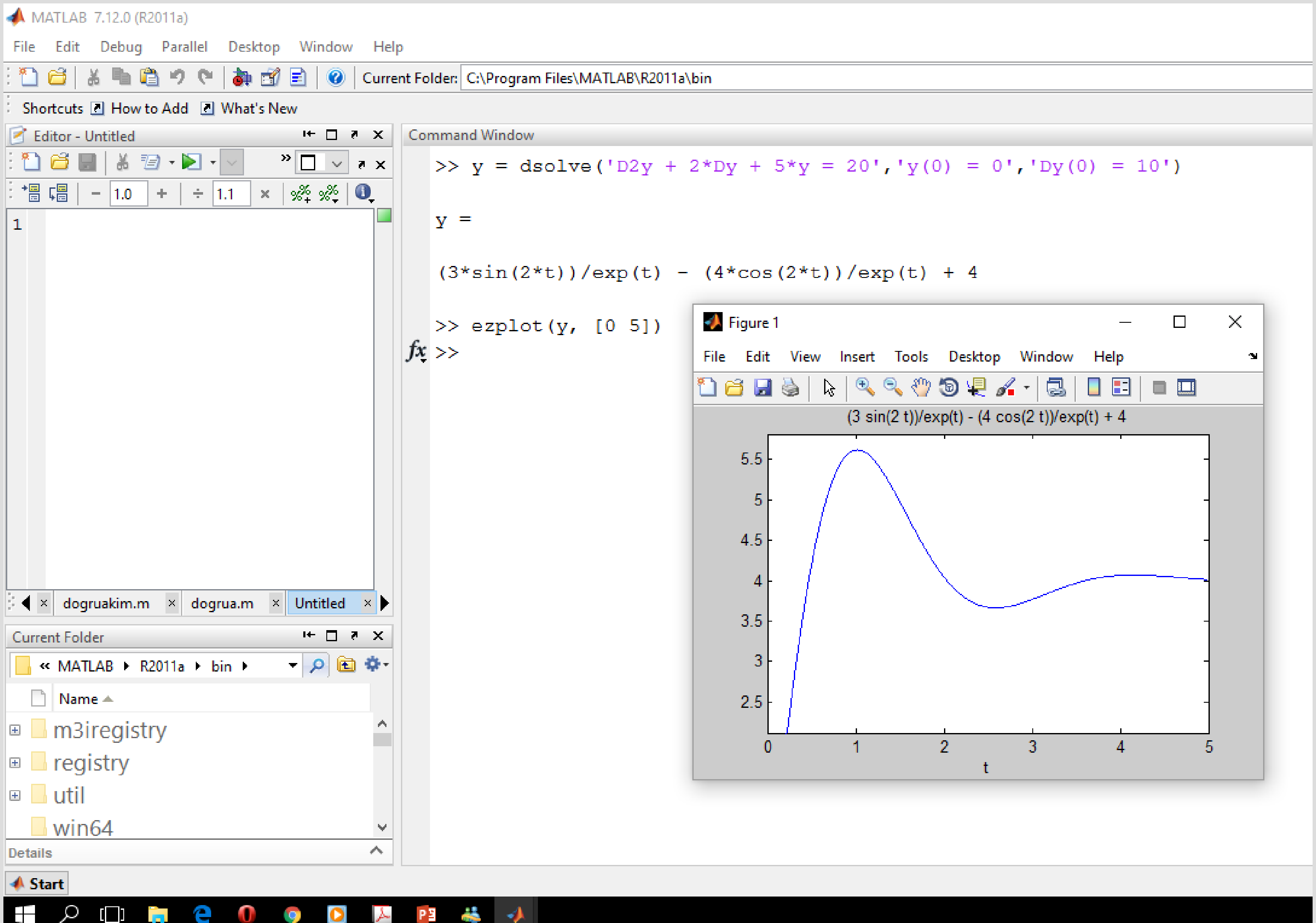


Ornek

$$\frac{dy}{dt} + 2y = 12\sin 4t \qquad y(0) = 10$$


MATLAB 7.12.0 (R2011a)
File Edit Debug Parallel Desktop Window Help
Current Folder: C:\Program Files\MATLAB\R2011a\bin
Shortcuts How to Add What's New
Editor - Untitled
Command Window
>> y=dsolve('Dy + 2*y = 12*sin(4*t)','y(0)=10')
y =
62/(5*exp(2*t)) - (12*cos(4*t))/5 + (6*sin(4*t))/5
>> ezplot(y, [0 8])
>> axis([0 8 -5 10])
>>
dograkim.m dogrua.m Untitled
Current Folder
MATLAB R2011a bin
Name
m3iregistry
registry
util
win64
Details
Start
Figure 1
File Edit View Insert Tools Desktop Window Help
62/(5 exp(2 t)) -...+ (6 sin(4 t))/5
10
5
0
-5
0 1 2 3 4 5 6 7 8
t

$$\frac{d^2y}{dt^2} + 2\frac{dy}{dt} + 5y = 20 \qquad y(0) = 0 \qquad y'(0) = 10$$



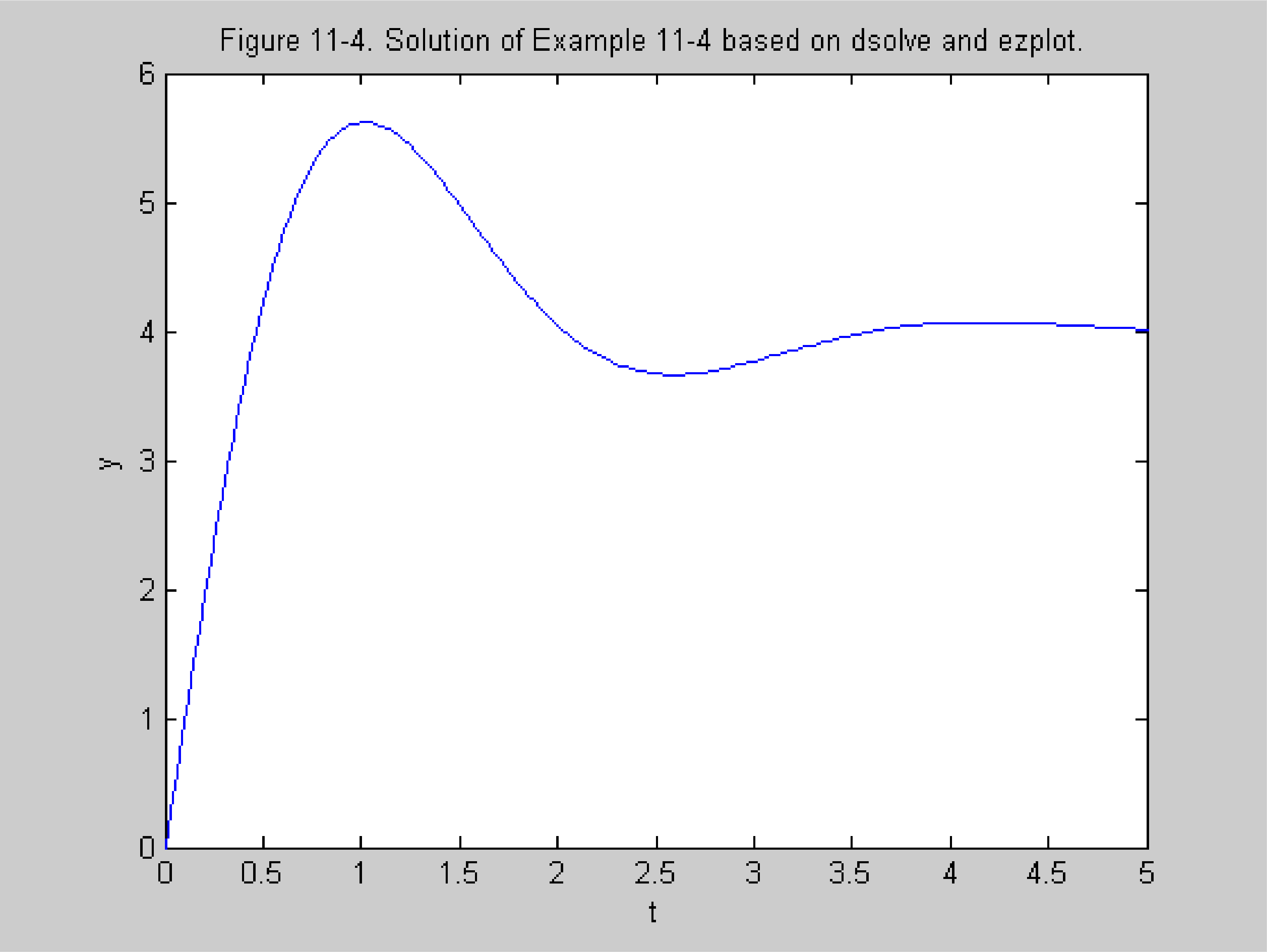
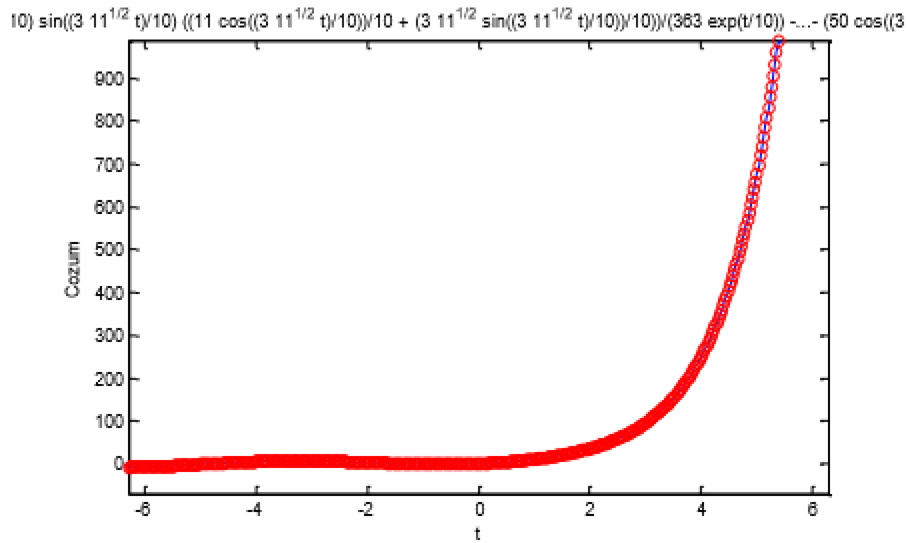


Figure 11-4. Solution of Example 11-4 based on dsolve and ezplot.
y
t
0
1
2
3
4
5
6
0.5
1.5
2.5
3.5
4.5

Örnek: $\boxed{\dfrac{d^2x}{dt^2} + 0.2\dfrac{dx}{dt} + x = 10e^{-t}}$ diferansiyel denklemini başlangıç koşullarına göre çözdürün ve grafiğini çizin.

x(0)=0 ; x'(0)=5

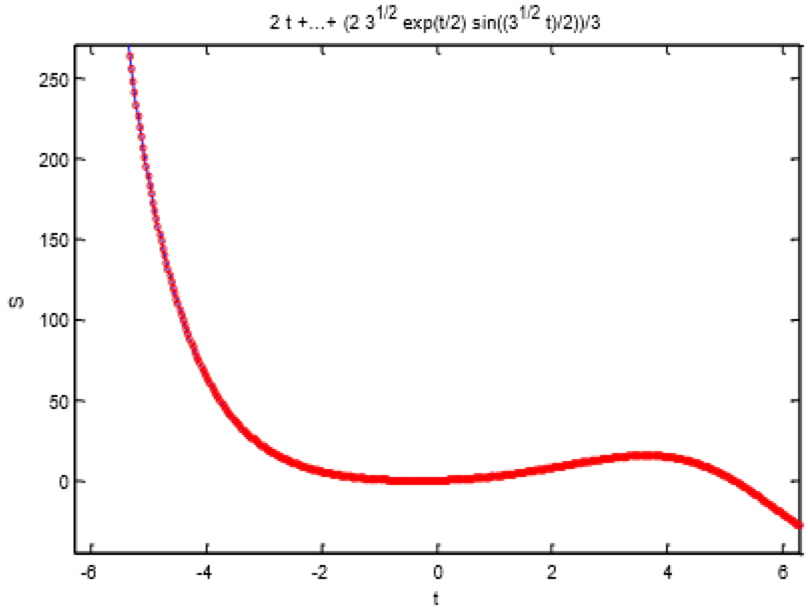
```
>>Cozum=dsolve('D2x+0.2*Dx+x=10*exp(t)','x(0)=0','
Dx(0)=5')

Cozum =

(500*11^(1/2)*exp((11*t)/10)*sin((3*11^(1/2)*t)/10
)*((11*cos((3*11^(1/2)*t)/10))/10 +
(3*11^(1/2)*sin((3*11^(1/2)*t)/10))/10))/(363*exp(t
/10)) -
(500*11^(1/2)*exp((11*t)/10)*cos((3*11^(1/2)*t)/1
0)*((11*sin((3*11^(1/2)*t)/10))/10 -
(3*11^(1/2)*cos((3*11^(1/2)*t)/10))/10))/(363*exp(
t/10)) - (50*cos((3*11^(1/2)*t)/10))/(11*exp(t/10))

>> ezplot(Cozum)
```



$x''' + 2x' = 2t$ Şeklindeki diferansiyel denklemi aşağıdaki sınır şartları ile dsolve komutu ile çözdürüp grafiğini çizelim;

x(0)=0 ; x'(0)=1 ; x''(0)=3

```
>>
S=dsolve('D3x+x=2*t','x(0)=0','Dx(0)=1','D2x(0)
=3')

S =

2*t + 4/(3*exp(t)) -
(4*exp(t/2)*cos((3^(1/2)*t)/2))/3 +
(2*3^(1/2)*exp(t/2)*sin((3^(1/2)*t)/2))/3

>> ezplot(S)
```



Cozmeyi deneyiniz

$$2\frac{dy}{dt} + y = 8$$

$$\frac{d^2y}{dt^2} + 5\frac{dy}{dt} + 6y = 3t$$

$$3\frac{d^4y}{dt^4} - \frac{d^3y}{dt^3} + 8\frac{dy}{dt} = \sin t$$

$$y'' - 3y' + 4y = 0$$

$$y^{(4)} + 8y^{(3)} + 24y'' + 32y' + 16y = 0$$

$$100\frac{d^2N}{dt^2} - 20*\frac{dN}{dt} + N = 0$$

$$y'' - y' - 2y = 4x^2$$

$$y'' - y' - 2y = e^{3x}$$

$$y'' - y' - 2y = \sin 2x$$

$$\frac{dI}{dt} + 20*I = 6\sin(2t)$$

$$xy' + y = xy^3$$

$$y' = \frac{y+x}{x}$$

$$\frac{dT}{dx} + kT = 100k$$

Adi diferansiyel denklemlerini çözünüz.

Diferansiyel denklemlerin sayısal çözümleri için bu denklemleri ikiye ayırmak mümkün.

- Doğrusal (Linear) Diferansiyel Denklemler
- Doğrusal olmayan (Nonlinear) Diferansiyel Denklemler

Doğrusal diferansiyel denklemlerin çözümü analitik olarak kolay yapılabilirken, doğrusal olmayanları yapmak zordur.

Bunun için Euler ve Runge-Kutta yaklaşımları gibi çeşitli sayısal yöntemler geliştirilmiştir

Matlab'ta diferansiyel denklemlerin çözümünde kullanılan "ode23" ve "ode45" adında iki adet Runge-Kutta fonksiyonu mevcuttur.

Bunlardan "ode23" fonksiyonu ikinci ve üçüncü dereceden Runge-Kutta integrasyon denklemlerini, ode45” fonksiyonu ise dördüncü ve beşinci dereceden Runge-Kutta integrasyon denklemlerini kullanır.

Bu denklemlerin çözümü için de m-file dosyasının oluşturulması gerekir.

NOT :Diferansiyel denklemlerin MATLAB’da sayısal çözümlerinde ode23, ode45, ode113, ode15s, ode23s, ode23t, ode23tb, ode15i olarak adlandırılan komutlar kullanılır. Bu komutlarda kullanilan cozum yontemleri farklilasir

.m dosyası hazırlayarak Diferansiyel Denklem Çözümü

- ode23 veya ode45 fonksiyonu aşağıdaki gibi kullanılır:
[t,y]=ode23('fonksiyon_adi',[t0 tson],y0,secenekler,p1,p2);

- fonksiyon_adı: diferansiyel denklemin tanımlı olduğu harici fonksiyon dosyasıdır. Adı tırnak içerisinde yazılmalıdır. bu dosya fonksiyon tanımlama kurallarına uygun olarak hazırlanmış olmalıdır. Tırnak içerine fonksiyonun ve dolayısıyla dosyanın adı yazılır. (.m) dosya uzantısının yazılmaması gerektiği unutulmamalıdır.

- Burada t değişkenine t0 dan başlayarak tson'a kadar iterasyonda kullanılan t değerleri atanır.

- y değişkeninde ise birinci sütuna t'ye (bağımsız değişkene) karşı y değerleri (bağımlı değişken değerleri) sırasıyla diğer sütunlara ise 1.,2., ... n. dereceden türevlerinin aldığı değerler atanır.

Steps for solving a single first-order ODE:

For the remainder of this section the independent variable is taken as t (time). This is done because in many applications time is the independent variable, and also to be consistent with the information in the **Help** menu of MATLAB.

Step 1: Write the problem in a standard form.

Write the equation in the form:

$$\frac{dy}{dt} = f(t, y) \quad \text{for} \quad t_0 \le t \le t_f, \text{ with } \quad y = y_0 \text{ at } t = t_0.$$

As shown above, three pieces of information are needed for solving a first order ODE: An equation that gives an expression for the derivative of y with respect to t, the interval of the independent variable, and the initial value of y. The solution is the value of y as a function of t between t_0 and t_f.

An example of a problem to solve is:

$$\frac{dy}{dt} = \frac{t^3 - 2y}{t} \quad \text{for} \quad 1 \le t \le 3 \quad \text{with} \quad y = 4.2 \quad \text{at } t = 1 .$$

<u>Step 2:</u> **Create a user-defined function (in a function file) or an anonymous function.**

The ODE to be solved has to be written as a user-defined function (in a function file) or as an anonymous function. Both calculate $\frac{dy}{dt}$ for given values of t and y.

For the example problem above, the user-defined function (which is saved as a separate file) is:

```
function dydt=ODEexp1(t,y)
dydt=(t^3-2*y)/t;
```

Step 3: **Select a method of solution.**

Select the numerical method that you would like MATLAB to use in the solution. Many numerical methods have been developed to solve first-order ODEs, and several of the methods are available as built-in functions in MATLAB. In a typical numerical method, the time interval is divided into small time steps. The solution starts at the known point y_0, and then by using one of the integration methods the value of y is calculated at each time step. Table 9-1 lists seven ODE solver commands, which are MATLAB built-in functions that can be used for solving a first-order ODE. A short description of each solver is included in the table.

Table 9-1: MATLAB ODE Solvers

ODE Solver Name	**Description**
`ode45`	For nonstiff problems, one-step solver, best to apply as a first try for most problems. Based on explicit Runge-Kutta method.
`ode23`	For nonstiff problems, one-step solver. Based on explicit Runge-Kutta method. Often quicker but less accurate than `ode45`.
`ode113`	For nonstiff problems, multistep solver.

Table 9-1: MATLAB ODE Solvers (Continued)

ODE Solver Name	Description
`ode15s`	For stiff problems, multistep solver. Use if `ode45` failed. Uses a variable order method.
`ode23s`	For stiff problems, one-step solver. Can solve some problems that `ode15s` cannot.
`ode23t`	For moderately stiff problems.
`ode23tb`	For stiff problems. Often more efficient than `ode15s`.

In general, the solvers can be divided into two groups according to their ability to solve stiff problems and according to whether they use on-step or multi-step methods. Stiff problems are ones that include fast and slowly changing components and require small time steps in their solution. One-step solvers use information from one point to obtain a solution at the next point. Multistep solvers use information from several previous points to find the solution at the next point. The details of the different methods are beyond the scope of this book.

It is impossible to know ahead of time which solver is the most appropriate for a specific problem. A suggestion is to first try `ode45`, which gives good results for many problems. If a solution is not obtained because the problem is stiff, trying the solver `ode15s` is suggested.

Step 4: **Solve the ODE.**

The form of the command that is used to solve an initial value ODE problem is the same for all the solvers and for all the equations that are solved. The form is:

```
[t,y] = solver_name(ODEfun,tspan,y0)
```

Additional information:

`solver_name` Is the name of the solver (numerical method) that is used (e.g. `ode45` or `ode23s`)

`ODEfun` The function from Step 2 that calculates $\frac{dy}{dt}$ for given values of t and y. If it was written as a user-defined function, the function handle is entered. If it was written as an anonymous function, the name of the anonymous function is entered. (See the example that follows.)

`tspan` A vector that specifies the interval of the solution. The vector must have at least two elements but can have more. If the vector has only two elements, the elements must be `[t0 tf]`, which are the initial and final points of the solution interval. The

tspan	Çözüm aralığını belirten bir vektördür. Vektör en az iki elemenae sahip olmalıdır, ancak daha fazlasına da sahip olabilir. Vektörün sadece iki elemanı varsa, elemanlar çözüm aralığının ilk ve son noktaları olan [t0 tf] olmalıdır. tspan, ancak, ilk ve son noktalar arasında ek noktalara sahip olabilir.
y0	y'nin başlangıç değeridir (aralığın ilk noktasında y değeri).
[t,y]	ODE'nin çözümü olan çıktıdır. t ve y sütun vektörleridir. İlk ve son noktalar, aralığın başlangıç ve bitiş noktalarıdır. Aradaki boşluk ve nokta sayısı, tspan giriş vektörüne bağlıdır. Eğer tspan'ın iki unsuru varsa (başlangıç ve bitiş noktaları), t ve y vektörleri çözücü tarafından hesaplanan her bir entegrasyon adımında çözümü içerir. Eğer tspan ikiden fazla noktaya sahipse (ilk ve sonuncu arasında ek puanlar), t ve y vektörleri sadece bu noktalarda çözüm içerir. Tspan'daki nokta sayısı, program tarafından çözüm için kullanılan zaman adımlarını etkilemez

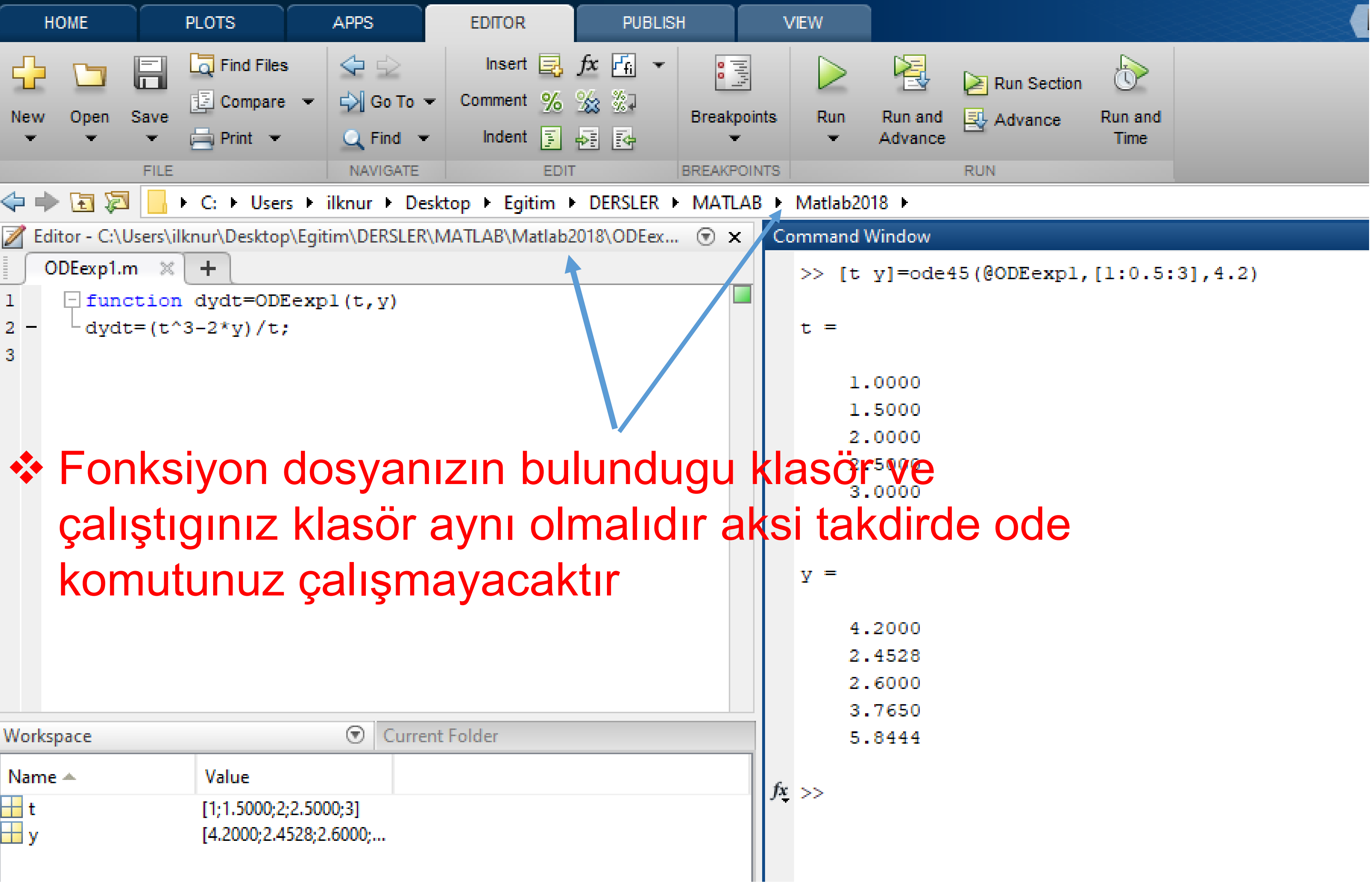
For example, consider the solution to the problem stated in Step 1:

$$\frac{dy}{dt} = \frac{t^3 - 2y}{t} \quad \text{for} \quad 1 \le t \le 3 \quad \text{with} \quad y = 4.2 \quad \text{at } t = 1,$$

<u>Step 2:</u> **Create a user-defined function (in a function file) or an anonymous function.**

The ODE to be solved has to be written as a user-defined function (in a function file) or as an anonymous function. Both calculate $\frac{dy}{dt}$ for given values of t and y. For the example problem above, the user-defined function (which is saved as a separate file) is:

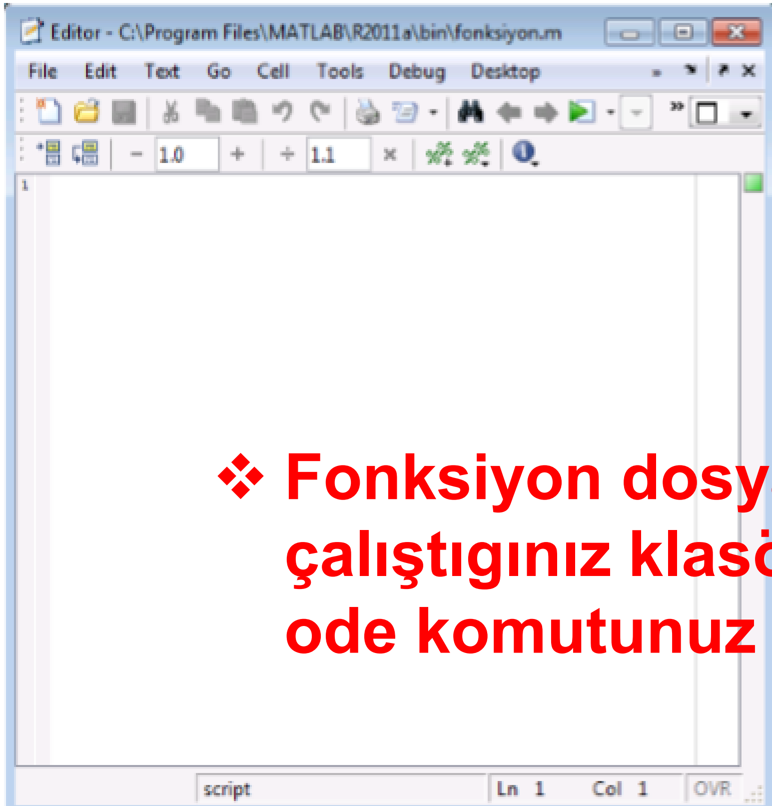
```
function dydt=ODEexp1(t,y)
dydt=(t^3-2*y)/t;
```



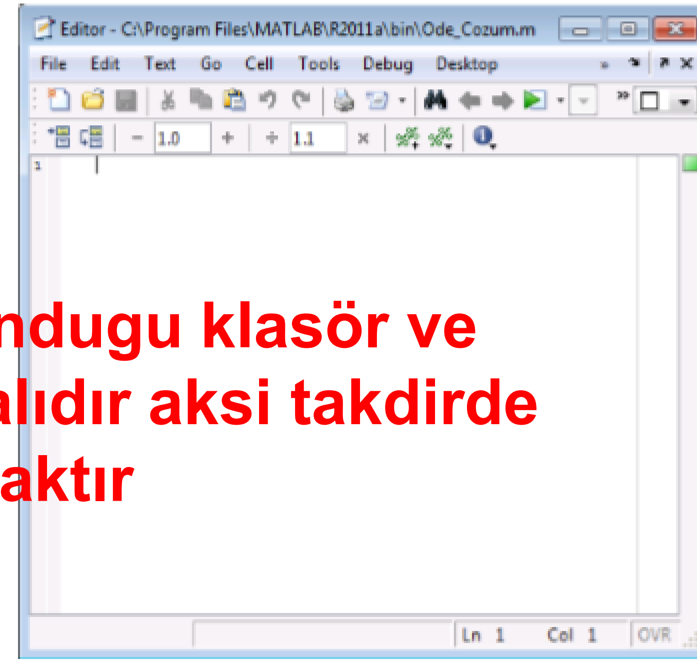
❖ Fonksiyon dosyanızın bulundugu klasör ve çalıştıgınız klasör aynı olmalıdır aksi takdirde ode komutunuz çalışmayacaktır

>> edit fonksiyon

Command window’a editör penceresinde girilecek foksiyonun adı yazılmalıdır ve kaydedilirken .m dosyası olarak kaydedilir.



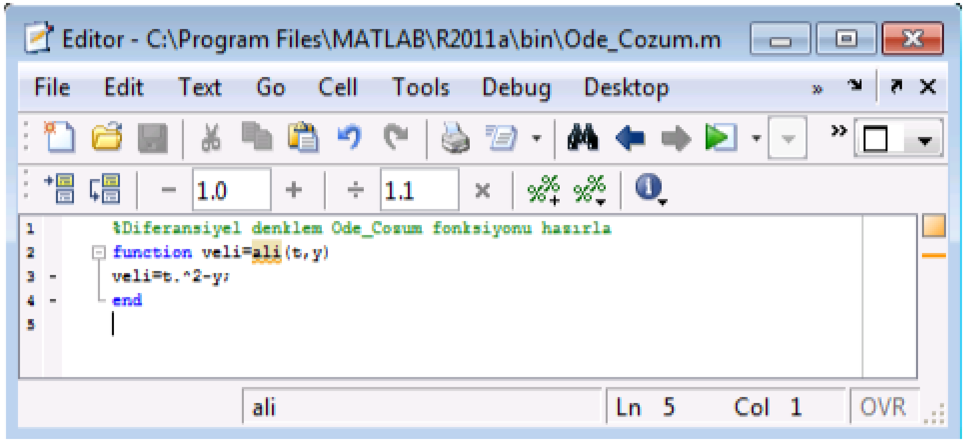
Mesela;
>> edit Ode_Cozum
olsun…



- **Fonksiyon dosyanızın bulundugu klasör ve çalıştıgınız klasör aynı olmalıdır aksi takdirde ode komutunuz çalışmayacaktır**

✤ Fonksiyon tanımlaması: $y' = \dfrac{dy}{dt} = t^2 - y$ şeklindeki adi diferansiyel denklemi editör penceresinde tanımlayalım;

```
%Diferansiyel denklem Ode_Cozum fonksiyonu hazırla
function veli=ali(t,y)
veli=t.^2-y;
end
```

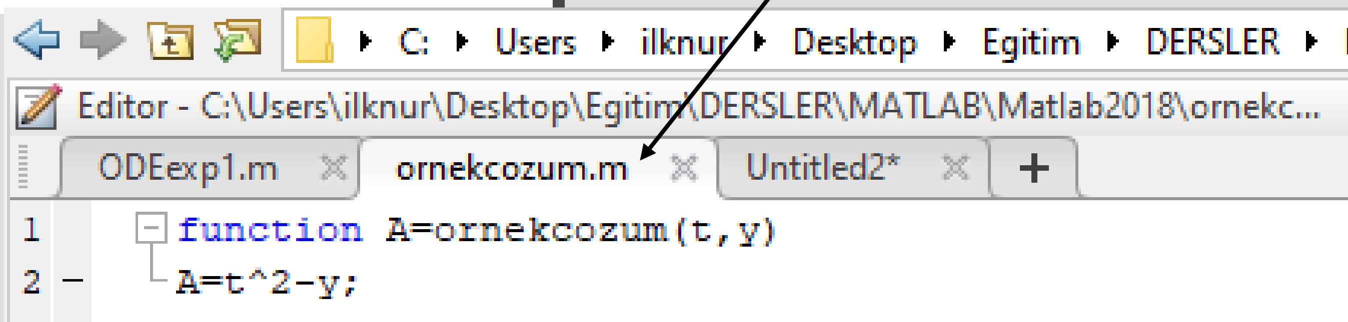


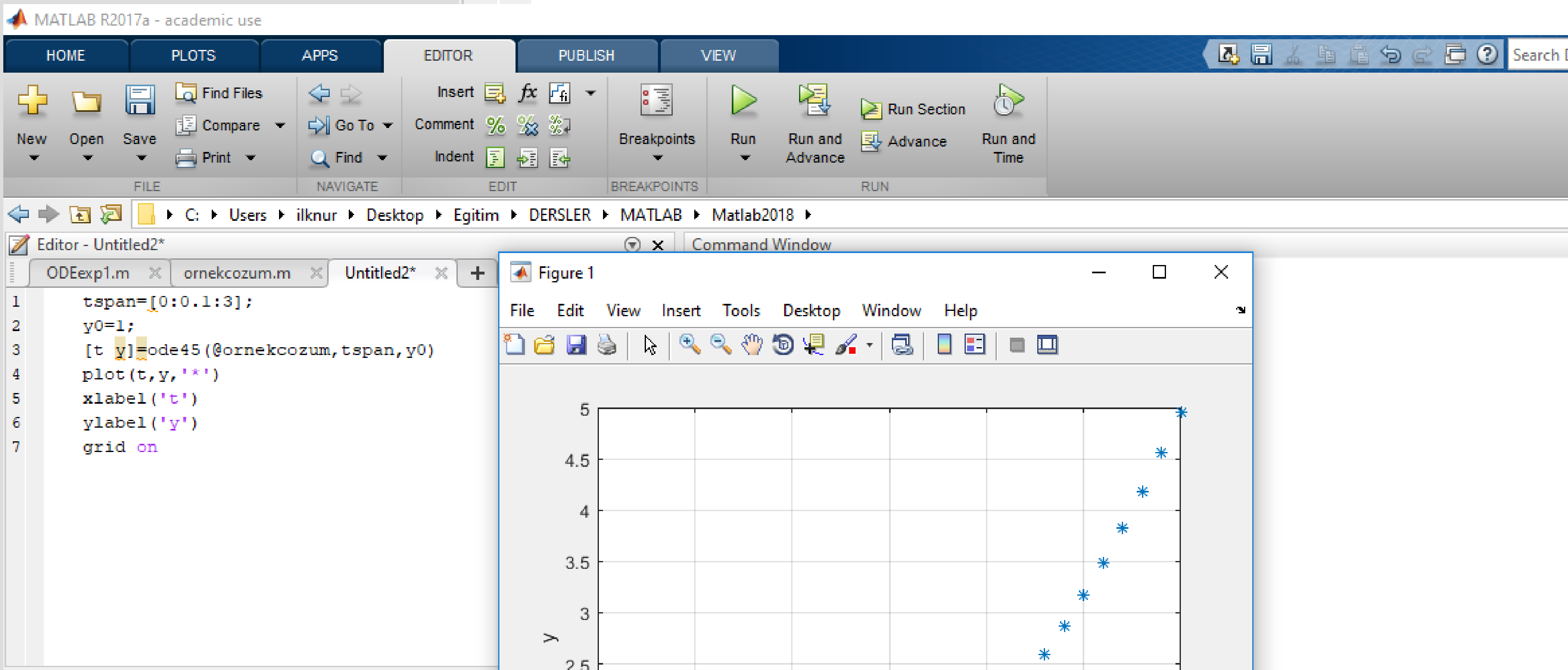
Ode_Cozum isimli fonksiyon, otomatik olarak kullanılmak üzere, save sekmesine tıklandığında kaydedilmiş olur.

$$y' = \frac{dy}{dt} = t^2 - y$$

şeklindeki adi diferansiyel denklemin çözümünü, y(0)=1 başlangıç koşuluyla, t(0)=0'dan tson=3'e kadarki değerleri için yapalım;

Once fonksiyon dosyasi olusturulur



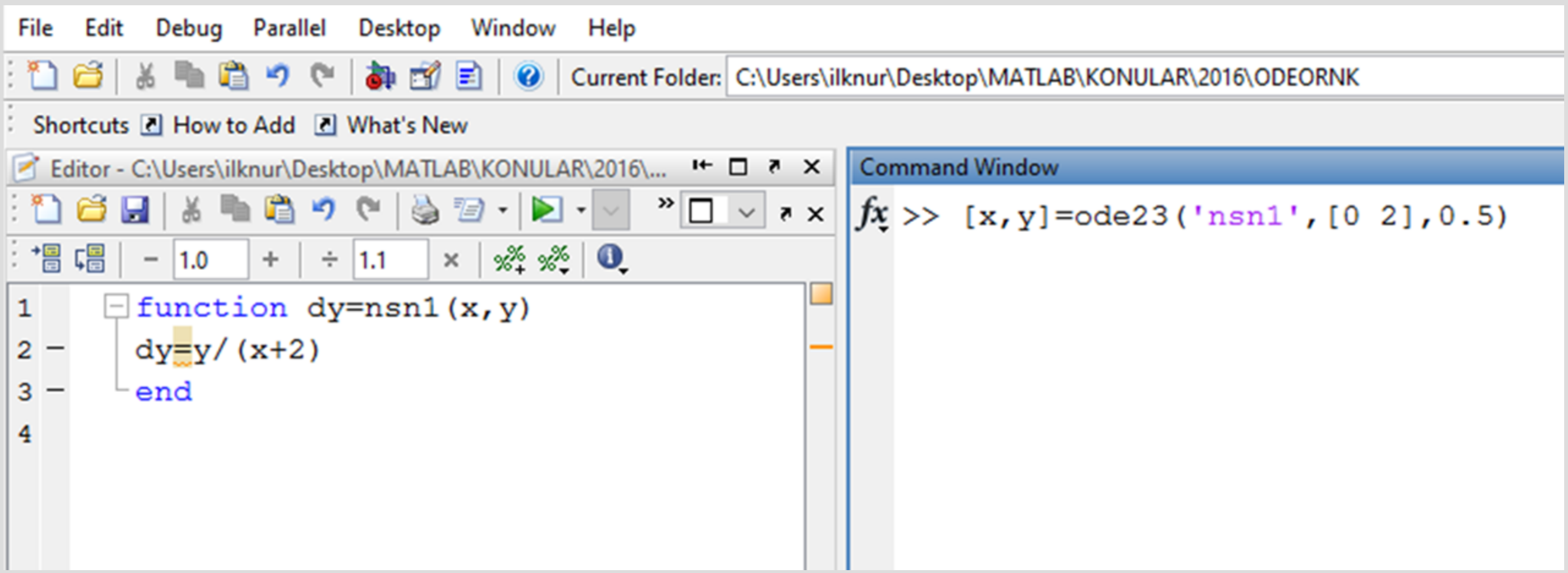


ODE23 ve ODE45 ILE ORNEK COZUMLER

$$y' = \frac{y}{x+2}$$

y(0)=0.5

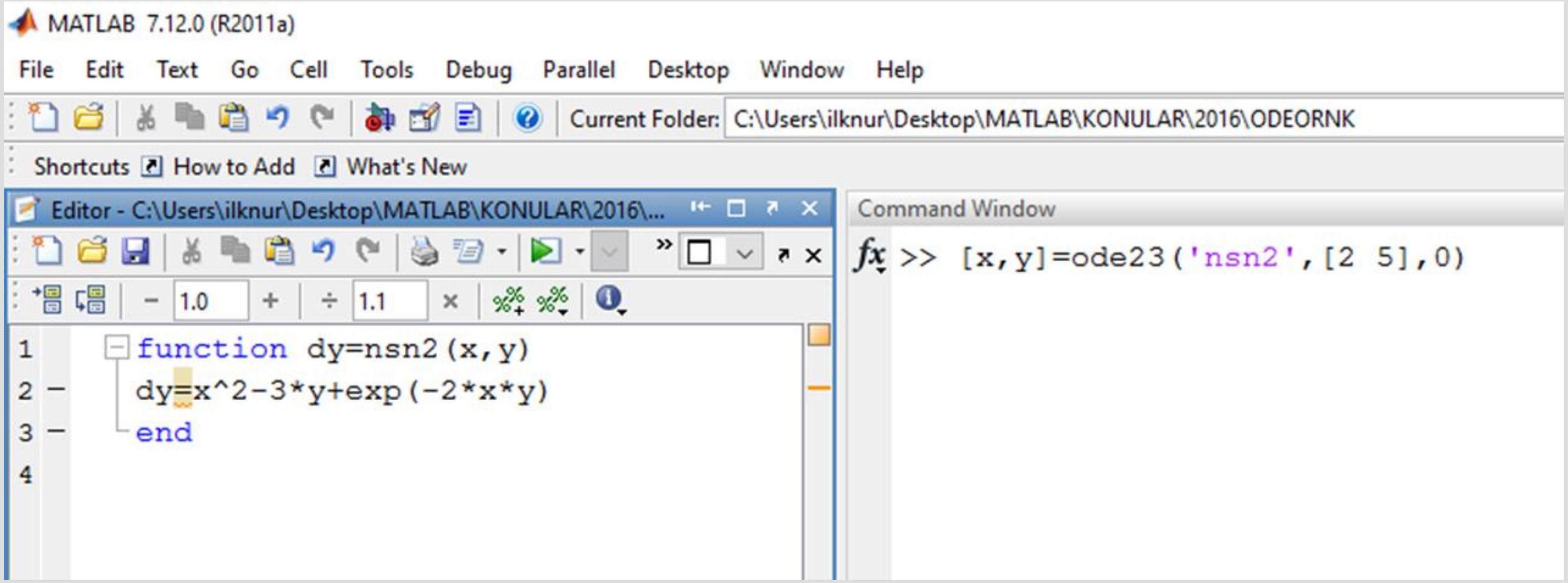
y[0 2]=?



$y' = x^2 - 3y + e^{-2xy}$

$y(0)=1$

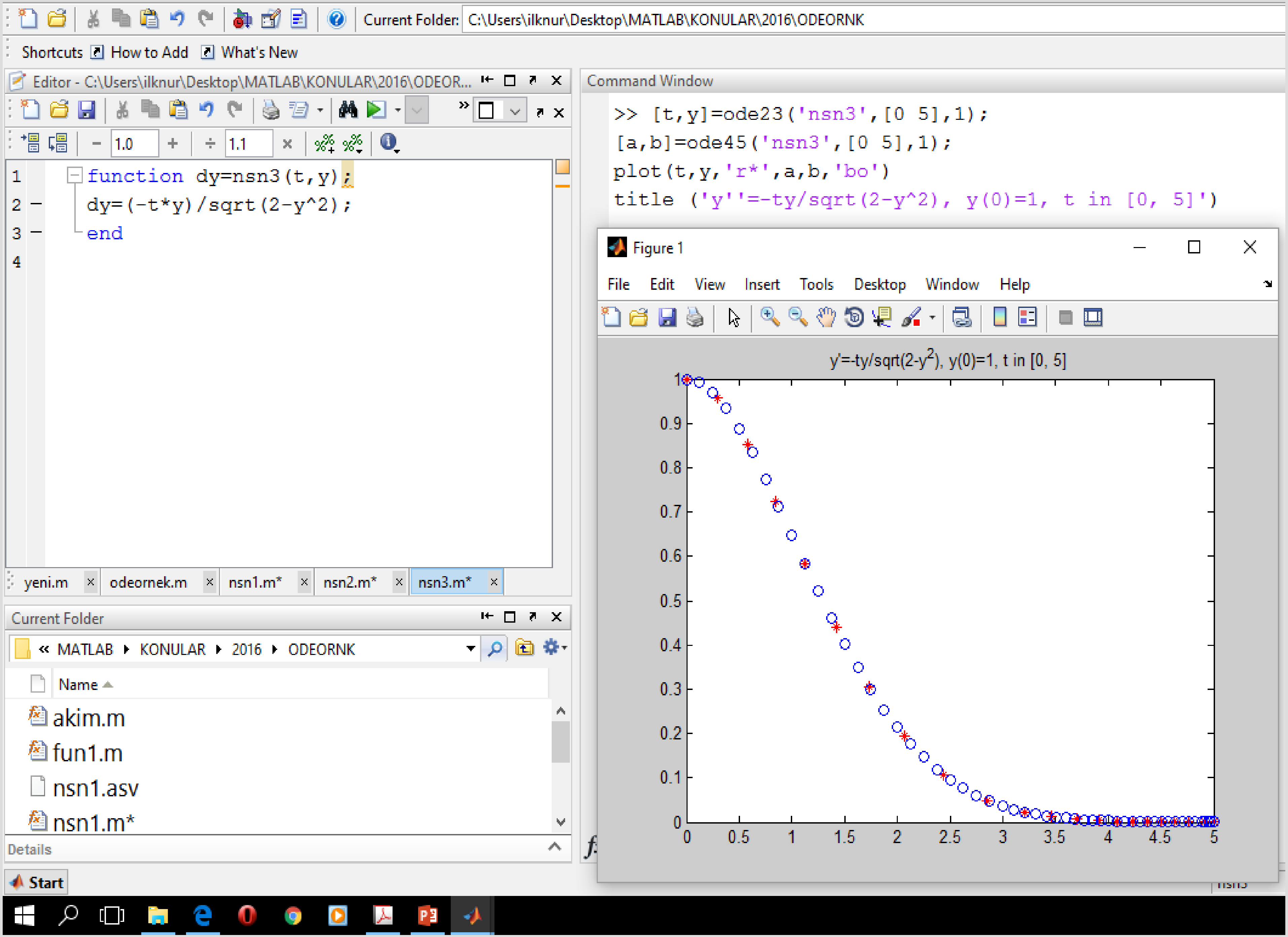
$y[2,5]=?$



$$y' = \frac{-ty}{\sqrt{2 - y^2}}$$

y(0)=1
to=0 ts=5

ode23 ve ode45 cozumunu karsilastiriniz (grafik ile)


Current Folder: C:\Users\ilknur\Desktop\MATLAB\KONULAR\2016\ODEORNK
Shortcuts
How to Add
What's New
Editor - C:\Users\ilknur\Desktop\MATLAB\KONULAR\2016\ODEOR...
function dy=nsn3(t,y);
dy=(-t*y)/sqrt(2-y^2);
end
yeni.m
odeornek.m
nsn1.m*
nsn2.m*
nsn3.m*
Current Folder
MATLAB
KONULAR
2016
ODEORNK
Name
akim.m
fun1.m
nsn1.asv
nsn1.m*
Details
Start
Command Window
>> [t,y]=ode23('nsn3',[0 5],1);
[a,b]=ode45('nsn3',[0 5],1);
plot(t,y,'r*',a,b,'bo')
title ('y''=-ty/sqrt(2-y^2), y(0)=1, t in [0, 5]')
Figure 1
File
Edit
View
Insert
Tools
Desktop
Window
Help
y'=-ty/sqrt(2-y^2), y(0)=1, t in [0, 5]
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5

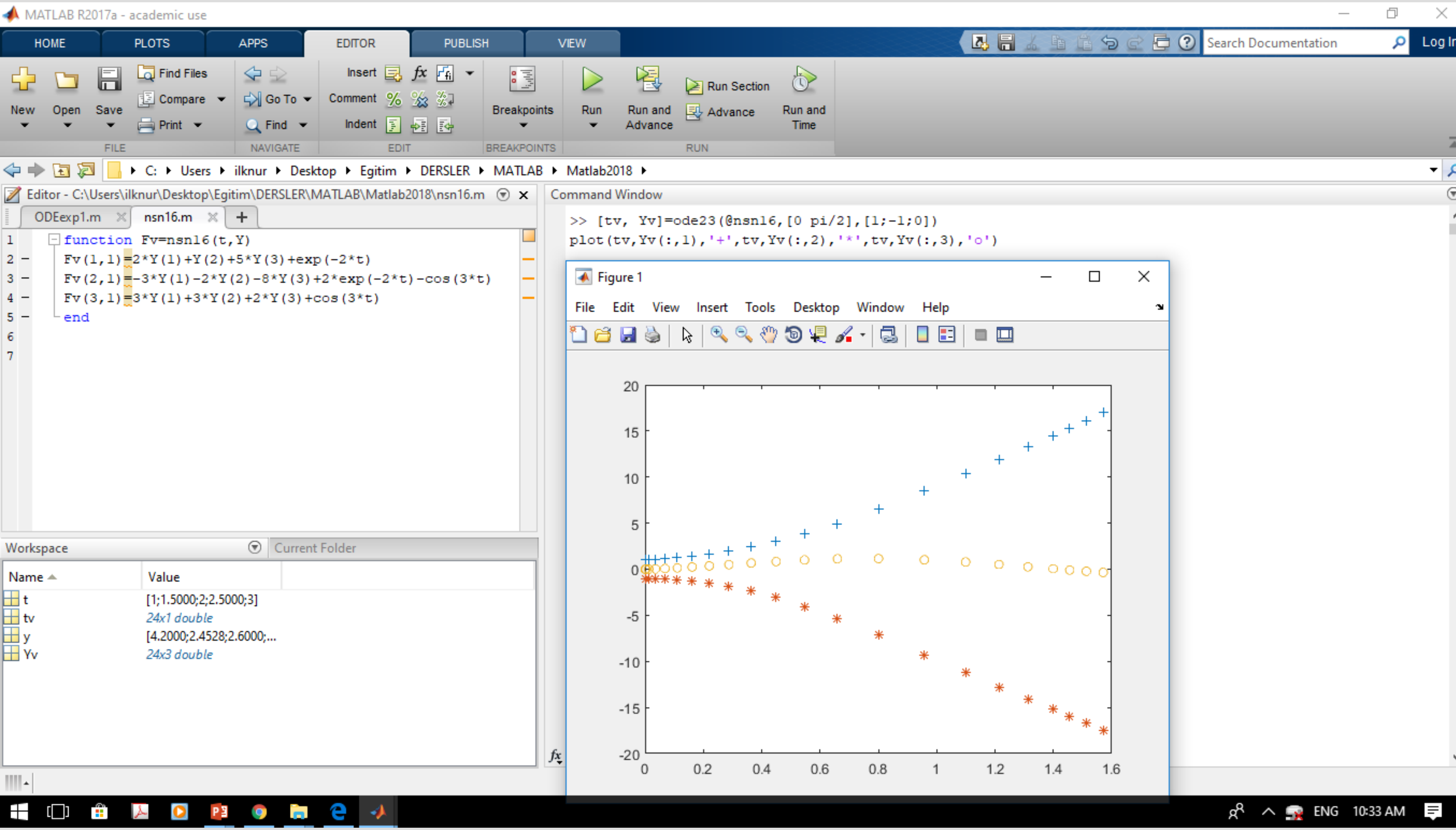
$y_1'=2y_1+y_2+5y_3+e^{-2t}$

$y_2'=-3y_1-2y_2-8y_3+2e^{-2t}-\cos(3t)$

$y_3'=3y_1+3y_2+2y_3+\cos(3t)$

$y_1(0)=1,\ y_2(0)=-1,\ y_3(0)=0 \qquad t\ \ [0,\ pi/2]$

ode23 veya ode45 kullanarak cozunuz ve grafik olarak gosteriniz

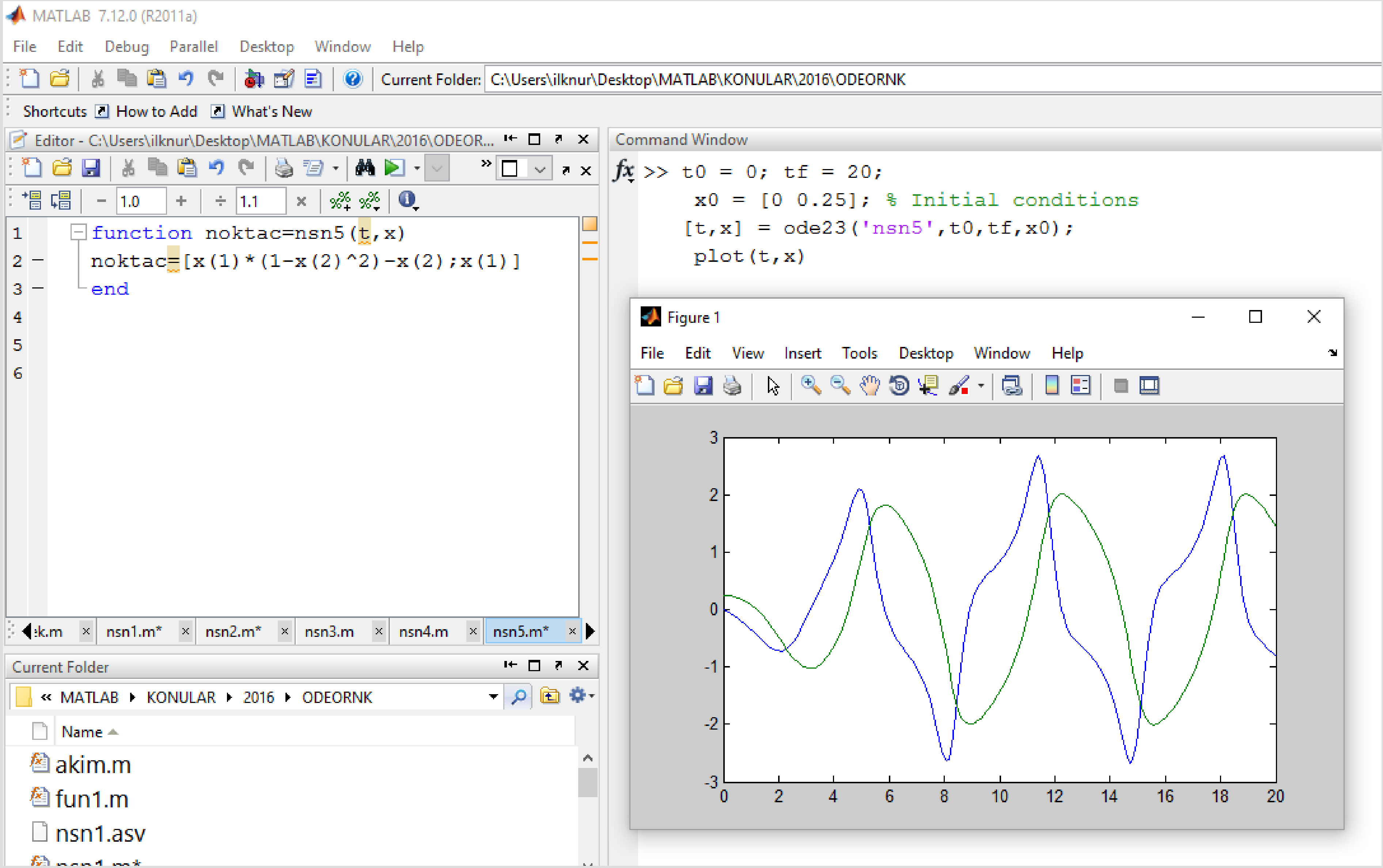

MATLAB R2017a - academic use
HOME
PLOTS
APPS
EDITOR
PUBLISH
VIEW
Search Documentation
Log In
Editor - C:\Users\ilknur\Desktop\Egitim\DERSLER\MATLAB\Matlab2018\nsn16.m
ODEexp1.m
nsn16.m
function Fv=nsn16(t,Y)
Fv(1,1)=2*Y(1)+Y(2)+5*Y(3)+exp(-2*t)
Fv(2,1)=-3*Y(1)-2*Y(2)-8*Y(3)+2*exp(-2*t)-cos(3*t)
Fv(3,1)=3*Y(1)+3*Y(2)+2*Y(3)+cos(3*t)
end
Command Window
>> [tv, Yv]=ode23(@nsn16,[0 pi/2],[1;-1;0])
plot(tv,Yv(:,1),'+',tv,Yv(:,2),'*',tv,Yv(:,3),'o')
Figure 1
Workspace
Current Folder
Name
Value
t
[1;1.5000;2;2.5000;3]
tv
24x1 double
y
[4.2000;2.4528;2.6000;...
Yv
24x3 double
20
15
10
5
0
-5
-10
-15
-20
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6

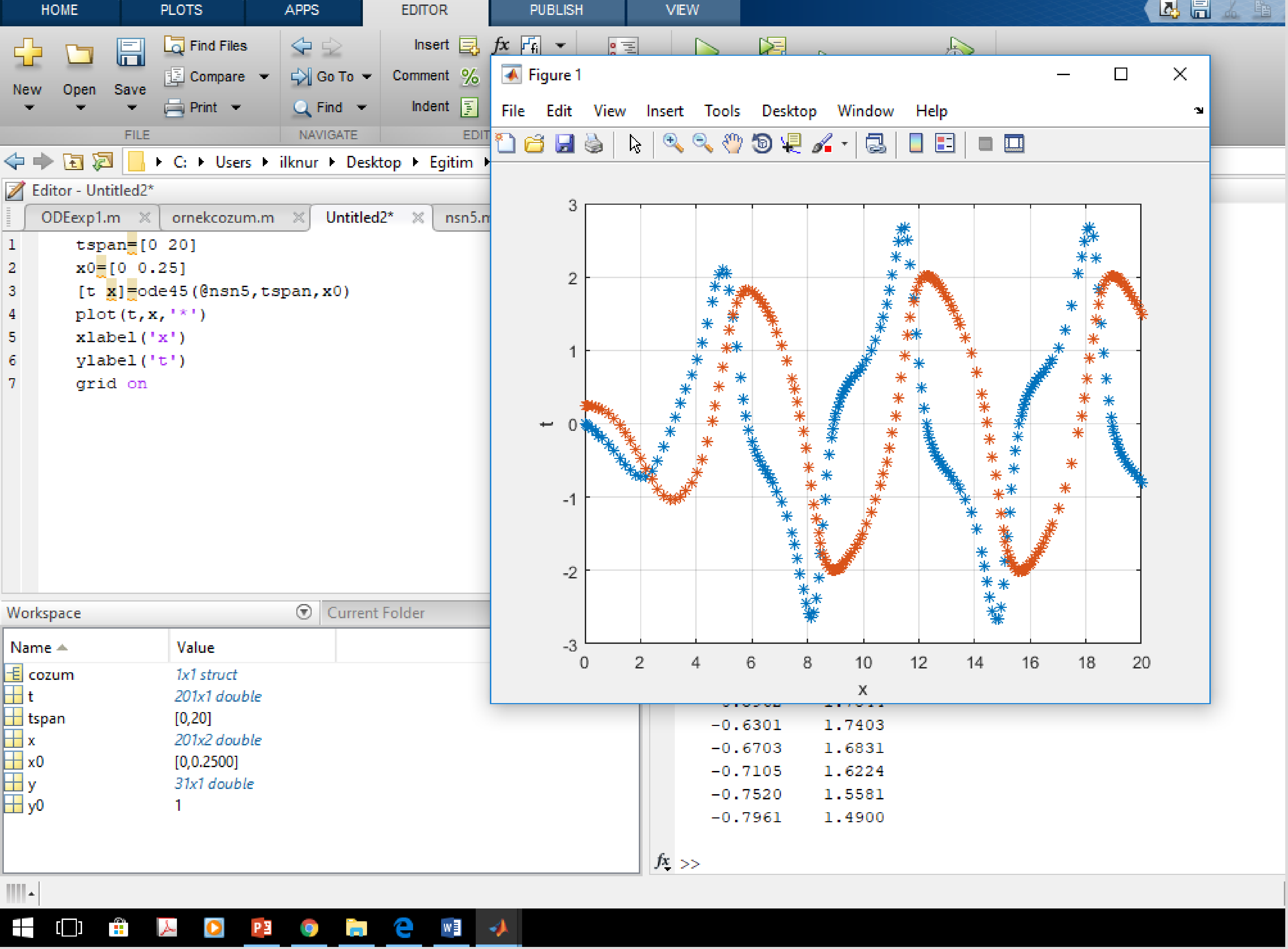
Van der Pol ($\ddot{x}+\left(x^2-1\right)\dot{x}+x=0$) Asagidaki sekilde yazilabilir. Bu denklemin cozumunu matlab ile yapiniz

$$\dot{x}_1 = x_1\left(1-x_2^2\right)-x_2$$
$$\dot{x}_2 = x_1$$

x [0 0.25]

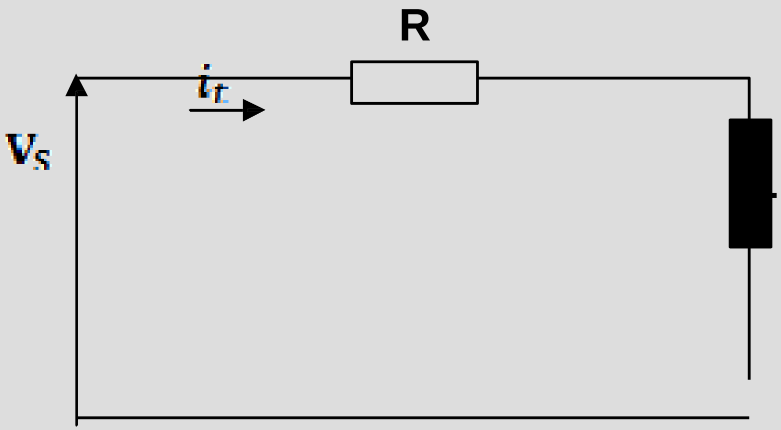
0<t<20


MATLAB 7.12.0 (R2011a)
File Edit Debug Parallel Desktop Window Help
Current Folder: C:\Users\ilknur\Desktop\MATLAB\KONULAR\2016\ODEORNK
Shortcuts How to Add What's New
Editor - C:\Users\ilknur\Desktop\MATLAB\KONULAR\2016\ODEOR...
function noktac=nsn5(t,x)
noktac=[x(1)*(1-x(2)^2)-x(2);x(1)]
end
:k.m nsn1.m* nsn2.m* nsn3.m nsn4.m nsn5.m*
Current Folder
« MATLAB KONULAR 2016 ODEORNK
Name
akim.m
fun1.m
nsn1.asv
Command Window
>> t0 = 0; tf = 20;
x0 = [0 0.25]; % Initial conditions
[t,x] = ode23('nsn5',t0,tf,x0);
plot(t,x)
Figure 1
File Edit View Insert Tools Desktop Window Help
3
2
1
0
-1
-2
-3
0 2 4 6 8 10 12 14 16 18 20



Editor - Untitled2*
ODEexp1.m
ornekcozum.m
Untitled2*
tspan=[0 20]
x0=[0 0.25]
[t x]=ode45(@nsn5,tspan,x0)
plot(t,x,'*')
xlabel('x')
ylabel('t')
grid on
Figure 1
x
t
Workspace
Name
Value
cozum 1x1 struct
t 201x1 double
tspan [0,20]
x 201x2 double
x0 [0,0.2500]
y 31x1 double
y0 1
-0.6301 1.7403
-0.6703 1.6831
-0.7105 1.6224
-0.7520 1.5581
-0.7961 1.4900

a-Alternatif Akımda:



Seri RL devresi

$V_s = 100sinwt$

R=0.4Ω

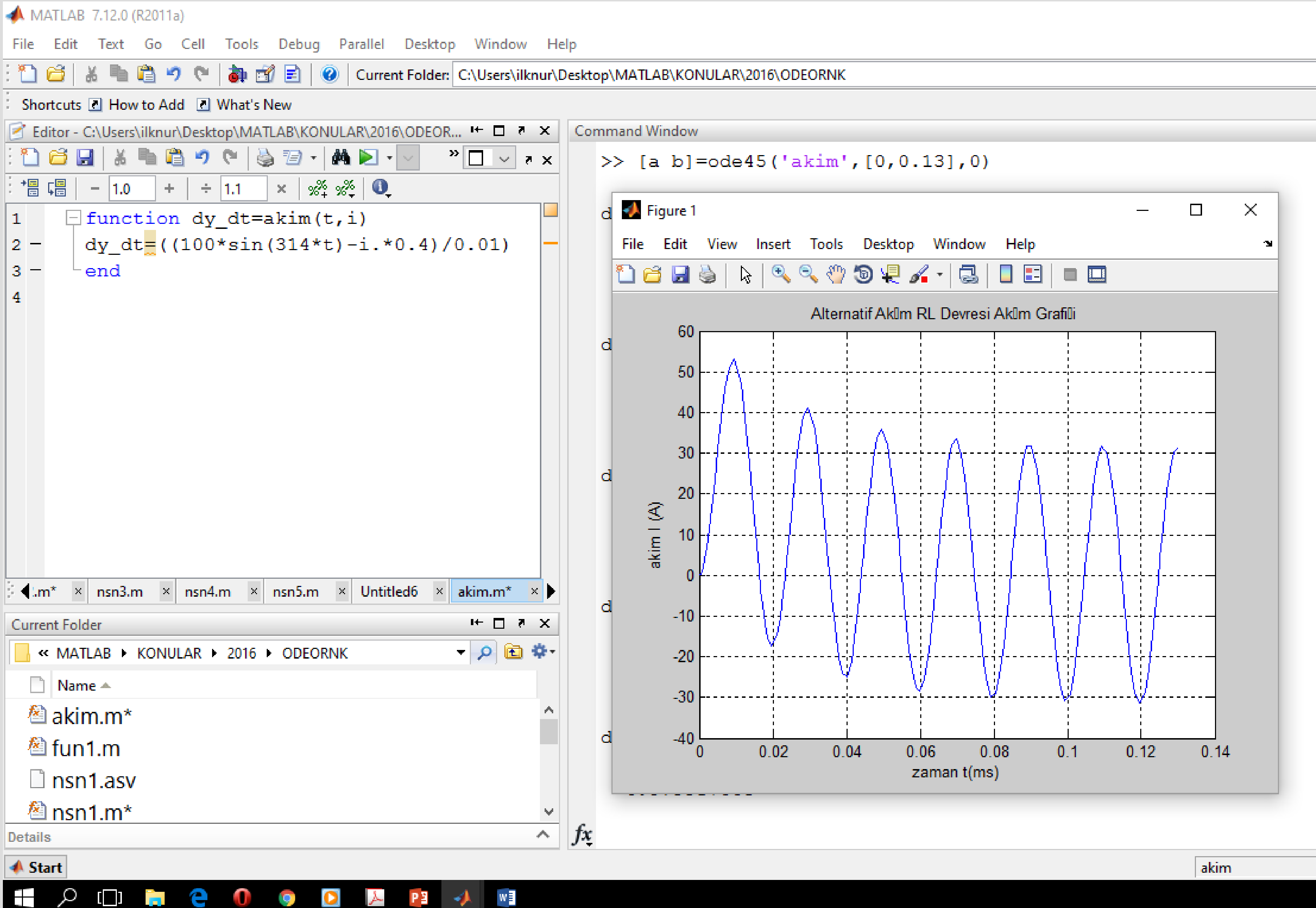
L=10 mH

f=50 Hz

i_t=?

$$V_s = R * i_t + L\frac{di}{dt}$$

w=2πf


MATLAB 7.12.0 (R2011a)
File Edit Text Go Cell Tools Debug Parallel Desktop Window Help
Current Folder: C:\Users\ilknur\Desktop\MATLAB\KONULAR\2016\ODEORNK
Shortcuts How to Add What's New
Editor - C:\Users\ilknur\Desktop\MATLAB\KONULAR\2016\ODEOR...
function dy_dt=akim(t,i)
dy_dt=((100*sin(314*t)-i.*0.4)/0.01)
end
.m* nsn3.m nsn4.m nsn5.m Untitled6 akim.m*
Current Folder
MATLAB KONULAR 2016 ODEORNK
Name
akim.m*
fun1.m
nsn1.asv
nsn1.m*
Details
Command Window
>> [a b]=ode45('akim',[0,0.13],0)
Figure 1
File Edit View Insert Tools Desktop Window Help
Alternatif Akım RL Devresi Akım Grafiği
akim I (A)
zaman t(ms)
Start
akim

Dogru akimda ise

$V_s = 100$

R=0.4Ω

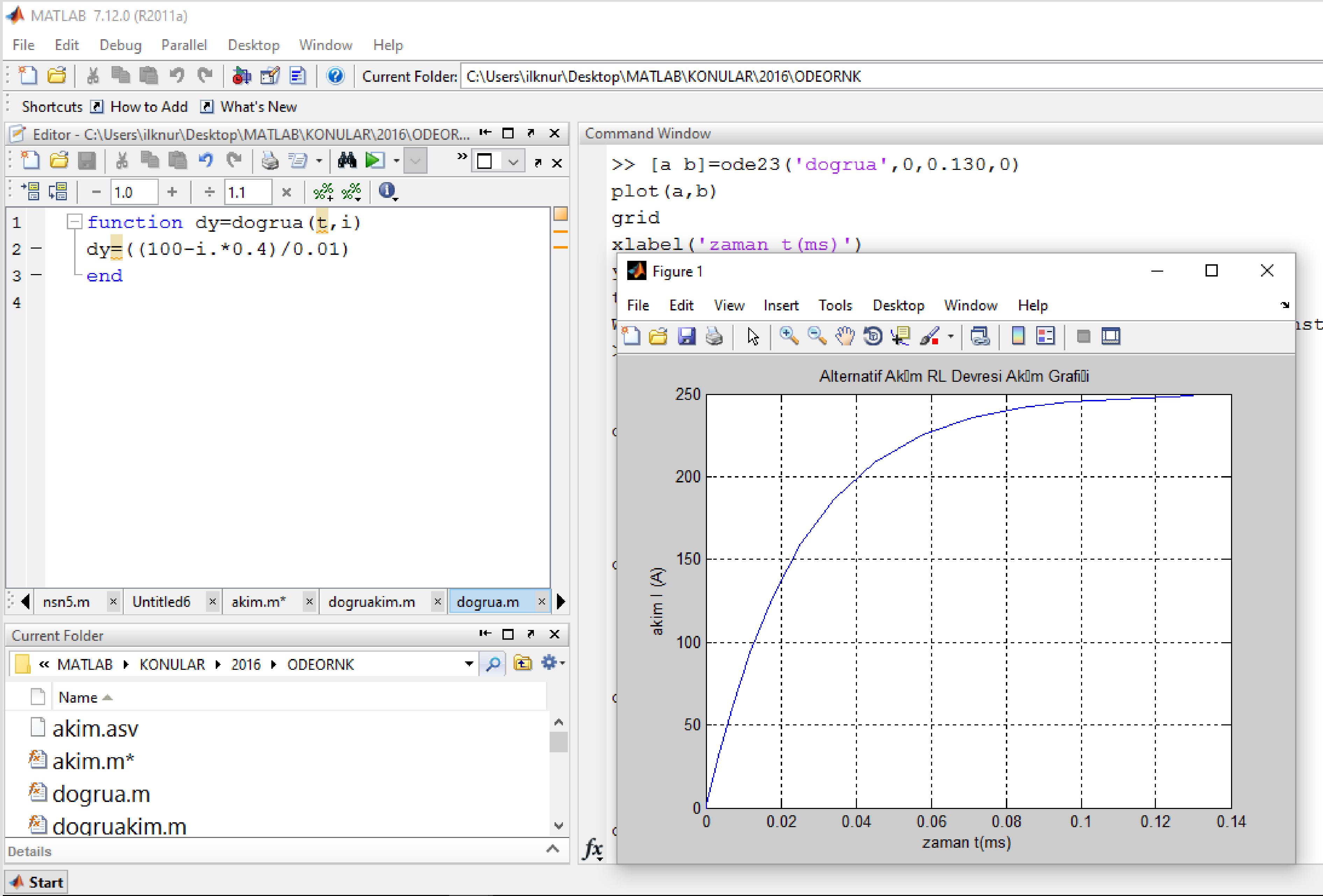
L=10 mH

f=50 Hz

i_t=?

$$V_s = R * i_t + L\frac{di}{dt}$$

$w=2\pi f$


MATLAB 7.12.0 (R2011a)
File Edit Debug Parallel Desktop Window Help
Current Folder: C:\Users\ilknur\Desktop\MATLAB\KONULAR\2016\ODEORNK
Shortcuts How to Add What's New
Editor - C:\Users\ilknur\Desktop\MATLAB\KONULAR\2016\ODEOR...
1.0
1.1
function dy=dogrua(t,i)
dy=((100-i.*0.4)/0.01)
end
nsn5.m
Untitled6
akim.m*
dogruakim.m
dogrua.m
Current Folder
« MATLAB ▸ KONULAR ▸ 2016 ▸ ODEORNK
Name
akim.asv
akim.m*
dogrua.m
dogruakim.m
Details
Start
Command Window
>> [a b]=ode23('dogrua',0,0.130,0)
plot(a,b)
grid
xlabel('zaman t(ms)')
Figure 1
File Edit View Insert Tools Desktop Window Help
Alternatif Ak□m RL Devresi Ak□m Grafi□i
akim I (A)
zaman t(ms)
0
50
100
150
200
250
0.02
0.04
0.06
0.08
0.1
0.12
0.14

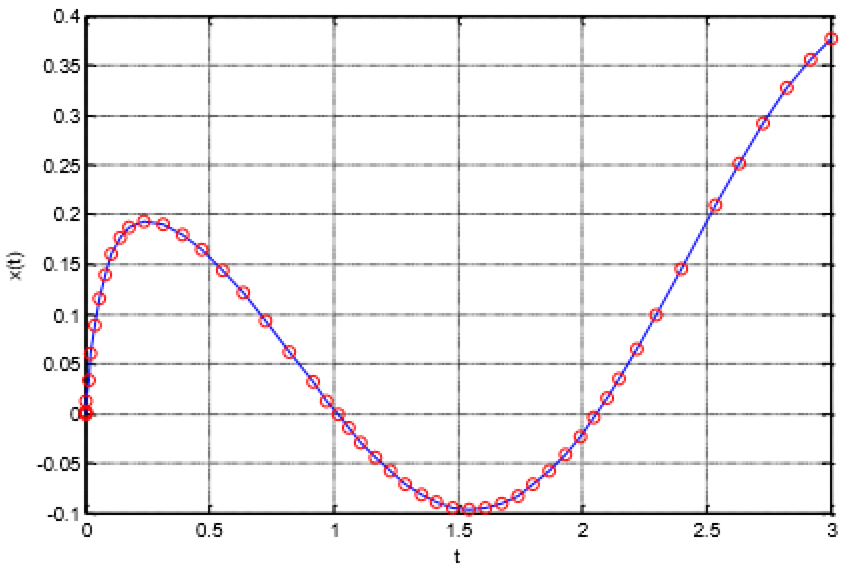
Örnek: $x' = \frac{dx(t)}{dt} = -15x(t) + 2\cos(2t) + t$ şeklindeki adi diferansiyel denklemin çözümünü, x(0)=0 başlangıç koşuluyla, t(0)=0'dan tson=3'e kadarki değerleri için yapalım;

```
% Diferansiyel denklem (Ornek.m)
function dx_dt=difdenk(t,x)
dx_dt=-15*x+3*cos(2*t)+t;
end
```

Fonksiyonun tanımlanması

Çözüm:

```
>> %Program dosyası (Ornek.m)
>> t0=0;
>> tson=3;
>> x0=0;
>> [t,x]=ode23('Ornek',[t0 tson],x0);
>> plot(t,x); xlabel('t'),ylabel('x(t)');grid on;
```



ode23 veya **ode45** komutlarıyla Diferansiyel Denklem Çözümü

Örnek: $$\frac{dx}{dt} = 2 \cdot x - x^2 - x \cdot y$$ $$\frac{dy}{dt} = x \cdot y - y$$

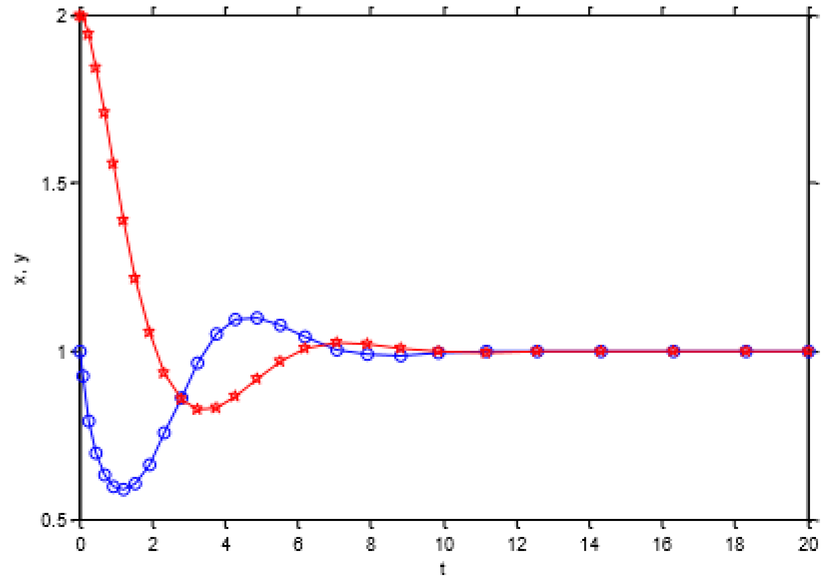
diferansiyel denklemini x(0)=1 ve y(0)=2 başlangıç koşullarına göre çözelim ve grafiğini çizelim.

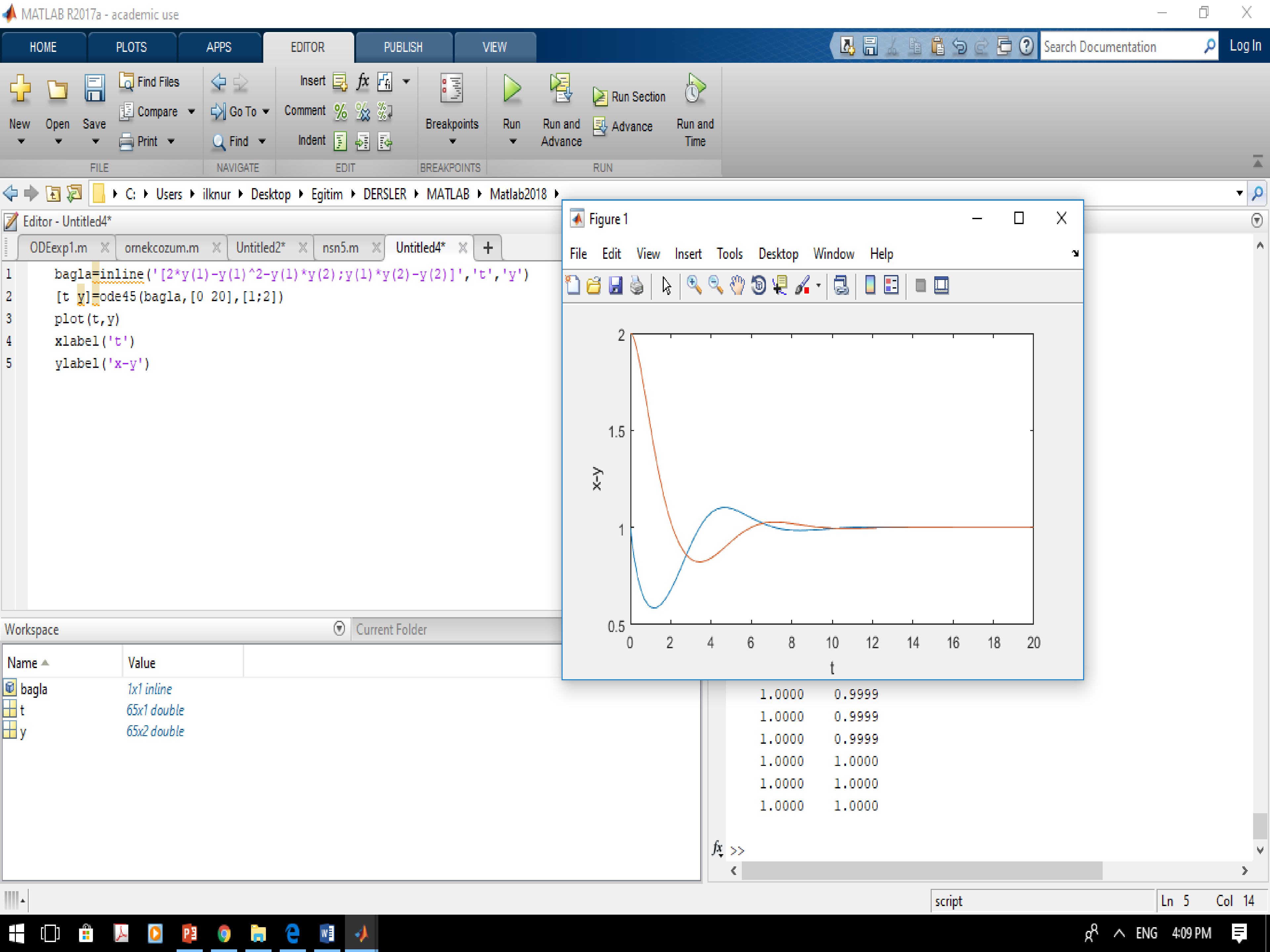
Çözüm için ilk önce x'i y(1), y'yi y(2) olarak tanımlayıp, iki denklemin eş zamanlı çözümü İçin **inline** komutu ile tek bir fonksiyon altında toplanması gerekir;

```
>> f=inline('[2*y(1)-y(1)^2-y(1)*y(2);y(1)*y(2)-y(2)]','t','y');
>> [t,y]=ode23(f,[0,20],[1;2])
>> plot(t,y)
```

veya

```
>> f=inline('[2*y(1)-y(1)^2-y(1)*y(2);y(1)*y(2)-y(2)]','t','y');
>> [t,y]=ode45(f,[0,20],[1;2])
>> plot(t,y)
```




MATLAB R2017a - academic use
HOME
PLOTS
APPS
EDITOR
PUBLISH
VIEW
Search Documentation
Log In
New
Open
Save
Find Files
Compare
Print
Go To
Find
Insert
Comment
Indent
Breakpoints
Run
Run and Advance
Run Section
Advance
Run and Time
FILE
NAVIGATE
EDIT
BREAKPOINTS
RUN
C: ▸ Users ▸ ilknur ▸ Desktop ▸ Egitim ▸ DERSLER ▸ MATLAB ▸ Matlab2018
Editor - Untitled4*
ODEexp1.m
ornekcozum.m
Untitled2*
nsn5.m
Untitled4*
bagla=inline('[2*y(1)-y(1)^2-y(1)*y(2);y(1)*y(2)-y(2)]','t','y')
[t y]=ode45(bagla,[0 20],[1;2])
plot(t,y)
xlabel('t')
ylabel('x-y')
Figure 1
File
Edit
View
Insert
Tools
Desktop
Window
Help
x-y
t
Workspace
Current Folder
Name
Value
bagla
1x1 inline
t
65x1 double
y
65x2 double
1.0000 0.9999
1.0000 0.9999
1.0000 0.9999
1.0000 1.0000
1.0000 1.0000
1.0000 1.0000
>>
script
Ln 5
Col 14
ENG
4:09 PM

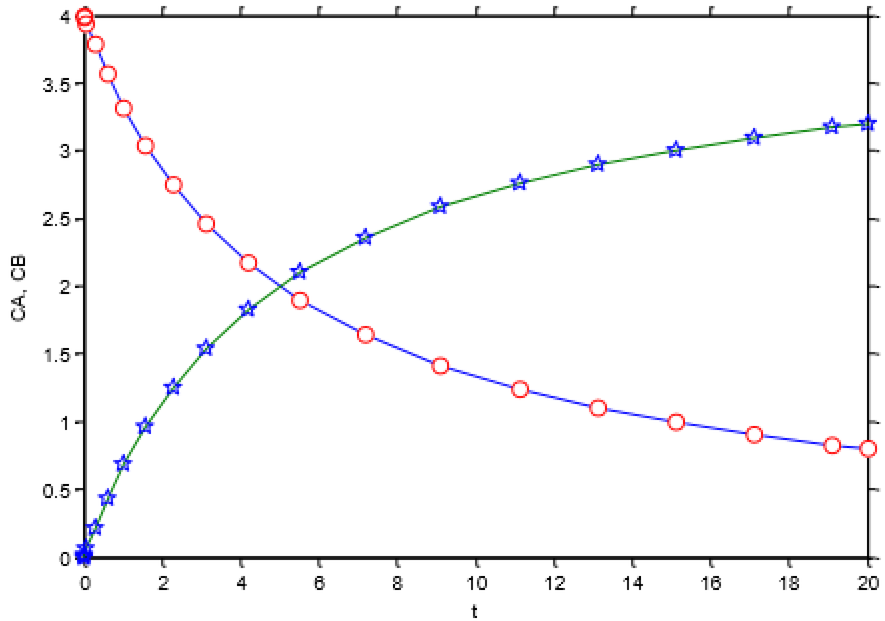
Örnek: $-r_A = -\dfrac{dC_A(t)}{dt} = kC_A^2$ şeklindeki 2. dereceden bir reaksiyon hız ifadesini, $C_A(0)=4$ mol/L ve $C_B=0$ mol/L başlangıç koşuluyla, t(0)=0'dan tson=20 dak'ya kadarki değişimi için çözelim; k=0.05

$$r_B = \frac{dC_B(t)}{dt} = k \cdot C_A^2$$

Çözüm için ilk önce C_A'yı y(1) ve C_B'yi y(2) olarak tanımlayıp, bir fonksiyonda tanımlayalım;

$$\frac{dy(1)}{dt} = -0.05 \cdot y(1)^2 \qquad \frac{dy(2)}{dt} = 0.05 \cdot y(1)^2$$

```
>> f=inline('[-0.05*y(1)^2;0.05*y(1)^2]','t','y');
>> [t,y]=ode23(f,[0,20],[4;0]);
>> plot(t,y)
```



Örnek: $A \xrightarrow{k_1} R \xrightarrow{k_2} S$ şeklindeki seri reaksiyonda, $C_A(0)$=4 mol/L ve C_R=C_S=0 mol/L başlangıç koşuluyla, her bir komponent için t(0)=0'dan tson=200 dak'ya kadarki değişimi için çözelim; k_1=0.05, k_2=0.012

Üç komponent için hız eşitlikleri ;

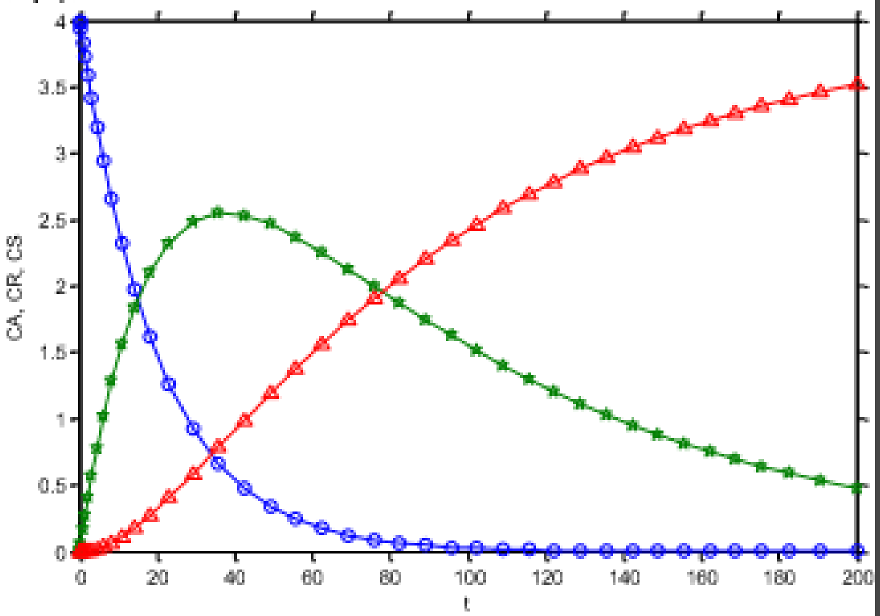
$$-\frac{dC_A}{dt} = k_1 C_A$$

$$\frac{dC_R}{dt} = k_1 C_A - k_2 C_R$$

$$\frac{dC_S}{dt} = k_2 C_R$$

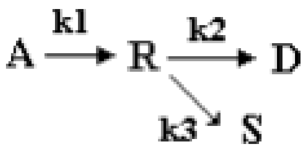
Çözüm için ilk önce C_A'yı y(1), C_R'yi y(2) ve C_S'yi y(3) olarak tanımlayıp, bir fonksiyonda tanımlayalım;

```
>> f=inline('[-0.05*y(1);0.05*y(1)-0.012*y(2);0.012*y(2)]','t','y');
>> [t,y]=ode23(f,[0,200],[4;0;0]);
>> plot(t,y)
```



Yukarıdaki diferansiyel denklemi başlangıç koşullarına göre 15'nci dakikaya kadar 30 adımda çözdürün ve grafiğini çizin.

6.



Seklindeki reaksiyonda her bir komponente ait hız ifadeleri aşağıda verilmiştir. 0'dan 80'nci dakikaya kadar, her komponentin konsantrasyon değişimini bulunuz ve grafik olarak gösteriniz. Başlangıçta CA konsantrasyonu 2 mol/lt'dir, CR ve CS yoktur. Sırasıyla k1=0.1 ; k2=0.05 ve k3=0.03 1/dk.'dir.

$$\frac{dCA}{dt} = -k1 \cdot CA \qquad \frac{dCD}{dt} = k2 \cdot CR$$

$$\frac{dCR}{dt} = k1 \cdot CA - (k2 + k3) \cdot CR \qquad \frac{dCS}{dt} = k3 \cdot CR$$

MATLAB 7.12.0 (R2011a)

File Edit Debug Parallel Desktop Window Help

Current Folder: C:\Users\ilknur\Desktop\Egitim\DERSLER\MATLAB\Matlab2017

Shortcuts How to Add What's New

Editor - C:\Users\ilknur\Desktop\Egitim\DERSLER\MATLAB\Matlab2...

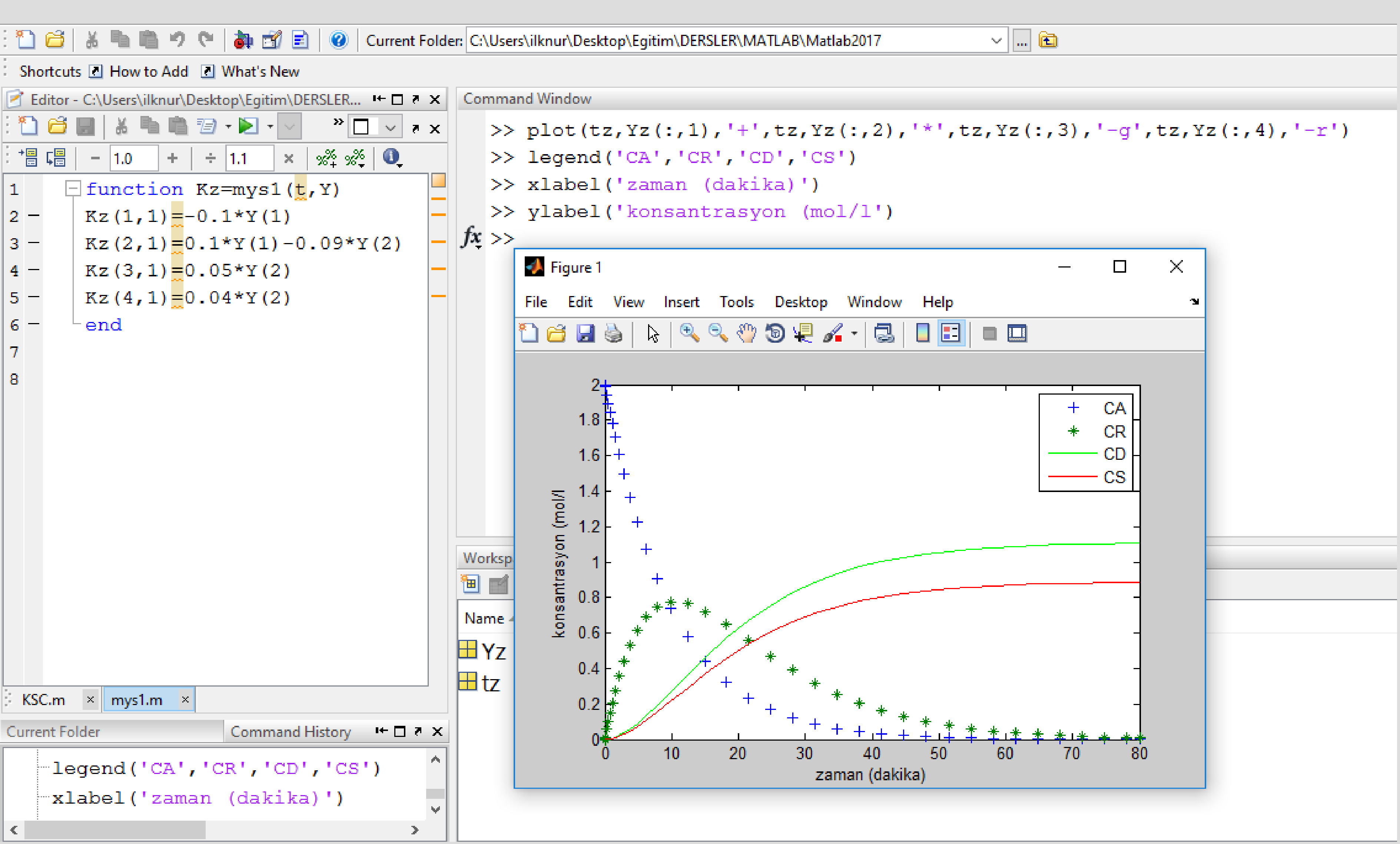
```
function Kz=mys1(t,Y)
Kz(1,1)=-0.1*Y(1)
Kz(2,1)=0.1*Y(1)-0.09*Y(2)
Kz(3,1)=0.05*Y(2)
Kz(4,1)=0.04*Y(2)
end
```

Command Window

```
>> [tz Yz]=ode23('mys1',[0 80],[2;0;0;0])

Kz =

   -0.2000


Kz =

   -0.2000
    0.2000
```

```
function Kz=mys1(t,Y)
Kz(1,1)=-0.1*Y(1)
Kz(2,1)=0.1*Y(1)-0.09*Y(2)
Kz(3,1)=0.05*Y(2)
Kz(4,1)=0.04*Y(2)
end
```

```
Yz =

    2.0000         0         0         0
    1.9999    0.0001    0.0000    0.0000
    1.9995    0.0005    0.0000    0.0000
    1.9975    0.0025    0.0000    0.0000
    1.9876    0.0124    0.0000    0.0000
    1.9413    0.0579    0.0004    0.0003
    1.8956    0.1019    0.0014    0.0011
    1.8454    0.1491    0.0031    0.0025
    1.7810    0.2077    0.0062    0.0050
    1.7014    0.2773    0.0118    0.0094
    1.6058    0.3564    0.0210    0.0168
    1.4941    0.4421    0.0354    0.0283
    1.3668    0.5303    0.0571    0.0457
```


Current Folder: C:\Users\ilknur\Desktop\Egitim\DERSLER\MATLAB\Matlab2017
Shortcuts How to Add What's New
Editor - C:\Users\ilknur\Desktop\Egitim\DERSLER...
function Kz=mys1(t,Y)
Kz(1,1)=-0.1*Y(1)
Kz(2,1)=0.1*Y(1)-0.09*Y(2)
Kz(3,1)=0.05*Y(2)
Kz(4,1)=0.04*Y(2)
end
KSC.m
mys1.m
Command Window
>> plot(tz,Yz(:,1),'+',tz,Yz(:,2),'*',tz,Yz(:,3),'-g',tz,Yz(:,4),'-r')
>> legend('CA','CR','CD','CS')
>> xlabel('zaman (dakika)')
>> ylabel('konsantrasyon (mol/l')
>>
Figure 1
File Edit View Insert Tools Desktop Window Help
CA
CR
CD
CS
konsantrasyon (mol/l
zaman (dakika)
Workspace
Name
Yz
tz
Current Folder
Command History
legend('CA','CR','CD','CS')
xlabel('zaman (dakika)')