7.12 EXAMPLES OF MATLAB APPLICATIONS

Sample Problem 7-5: Exponential growth and decay

A model for exponential growth or decay of a quantity is given by

$$A(t) = A_0 e^{kt}$$

where $A(t)$ and A_0 are the quantity at time t and time 0, respectively, and k is a constant unique to the specific application.

Write a user-defined function that uses this model to predict the quantity $A(t)$ at time t from knowledge of A_0 and $A(t_1)$ at some other time t_1. For function name and arguments, use `At = expGD(A0,At1,t1,t)`, where the output argument `At` corresponds to $A(t)$, and for input arguments, use `A0,At1,t1,t,` corresponding to A_0, $A(t_1)$, t_1, and t, respectively.

Use the function file in the Command Window for the following two cases:

(*a*) The population of Mexico was 67 million in the year 1980 and 79 million in 1986. Estimate the population in 2000.

(*b*) The half-life of a radioactive material is 5.8 years. How much of a 7-gram sample will be left after 30 years?

Solution

To use the exponential growth model, the value of the constant k has to be determined first by solving for k in terms of A_0, $A(t_1)$, and t_1:

$$k = \frac{1}{t_1}\ln\frac{A(t_1)}{A_0}$$

Once k is known, the model can be used to estimate the population at any time.

The user-defined function that solves the problem is:

```
function At=expGD(A0,At1,t1,t)                Function definition line.
% expGD calculates exponential growth and decay
% Input arguments are:
% A0: Quantity at time zero.
% At1: Quantity at time t1.
% t1: The time t1.
% t: time t.
% Output argument is:
% At: Quantity at time t.
k=log(At1/A0)/t1;                             Determination of k.
At=A0*exp(k*t);                               Determination of A(t).
                                              (Assignment of value to output variable.)
```

Once the function is saved, it is used in the Command Window to solve the two cases. For case *a*) $A_0 = 67$, $A(t_1) = 79$, $t_1 = 6$, and $t = 20$:

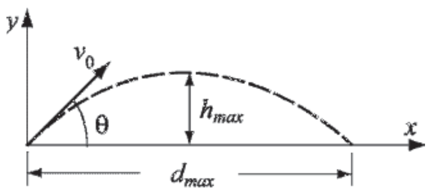
```
>> expGD(67,79,6,20)
ans =
        116.03          Estimation of the population in the year 2000.
```

For case *b*) $A_0 = 7$, $A(t_1) = 3.5$ (since t_1 corresponds to the half-life, which is the time required for the material to decay to half of its initial quantity), $t_1 = 5.8$, and $t = 30$.

```
>> expGD(7,3.5,5.8,30)
ans =
           0.19          The amount of material after 30 years.
```

Sample Problem 7-6: Motion of a projectile

Create a function file that calculates the trajectory of a projectile. The inputs to the function are the initial velocity and the angle at which the projectile is fired. The outputs from the function are the maximum height and distance. In addition, the function generates a plot of the trajectory. Use the function to calculate the trajectory of a projectile that is fired at a velocity of 230 m/s at an angle of 39°.



Solution

The motion of a projectile can be analyzed by considering the horizontal and vertical components. The initial velocity v_0 can be resolved into horizontal and vertical components

$$v_{0x} = v_0 \cos(\theta) \quad \text{and} \quad v_{0y} = v_0 \sin(\theta)$$

In the vertical direction the velocity and position of the projectile are given by:

$$v_y = v_0 - gt \quad \text{and} \quad y = v_{0y}t - \frac{1}{2}gt$$

The time it takes the projectile to reach the highest point ($v_y = 0$) and the corresponding height are given by:

$$t_{hmax} = \frac{v_{0y}}{g} \quad \text{and} \quad h_{hmax} = \frac{v_{0y}^2}{2g}$$

The total flying time is twice the time it takes the projectile to reach the highest point, $t_{tot} = 2t_{hmax}$. In the horizontal direction the velocity is constant, and the position of the projectile is given by:

$$x = v_{0x}t$$

In MATLAB notation the function name and arguments are entered as `[hmax,dmax] = trajectory(v0,theta)`. The function file is:

```
function [hmax,dmax]=trajectory(v0,theta)      Function definition line.
% trajectory calculates the max height and distance of a
projectile, and makes a plot of the trajectory.
% Input arguments are:
% v0: initial velocity in (m/s).
% theta: angle in degrees.
% Output arguments are:
% hmax: maximum height in (m).
% dmax: maximum distance in (m).
% The function creates also a plot of the trajectory.
g=9.81;
v0x=v0*cos(theta*pi/180);
v0y=v0*sin(theta*pi/180);
thmax=v0y/g;
hmax=v0y^2/(2*g);
ttot=2*thmax;
dmax=v0x*ttot;
% Creating a trajectory plot
tplot=linspace(0,ttot,200);   Creating a time vector with 200 elements.
x=v0x*tplot;                    Calculating the x and y coordi-
y=v0y*tplot-0.5*g*tplot.^2;     nates of the projectile at each time.
plot(x,y)                       Note the element-by-element multiplication.
xlabel('DISTANCE (m)')
ylabel('HEIGHT (m)')
title('PROJECTILE''S TRAJECTORY')
```

After the function is saved, it is used in the Command Window for a projectile that is fired at a velocity of 230 m/s and an angle of 39°.

```
>> [h d]=trajectory(230,39)
h =
  1.0678e+003
d =
  5.2746e+003
```