MAT 343 Laboratory 2
Linear transformations and computer animation in MATLAB

Bruno D. Welfert

Arizona State University
School of Mathematical & Statistical Sciences

Goals

In this laboratory session you learn how to

- Plot in MATLAB
- Implement linear transformations in MATLAB (rotation, reflection, scaling)
- Perform simple computer animation in MATLAB
- Use homogeneous coordinates to create more advanced computer animations
- Create AVI movie in MATLAB

Plotting in MATLAB

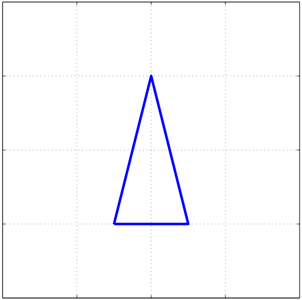
Use plot to graph data/functions

```
x = 0:0.2:10; % 51 values 0,0.2,0.4,0.6,...,10.0
f =@(x)x.*sin(x)./(1+x); % inline function
    % note .* and ./ for componentwise * and /
y = f(x); % evaluate f at list x
plot(x,y); % graph y data vs x data
plot(x,y,'o'); % use marker o without connecting points
z = sin(x); % evaluate sin at x values
plot(x,y,'o',x,z); % superpose 2 plots
T = [x;y]; % 2x51 matrix with x in row 1 and y in row 2
plot(T(1,:),T(2,:)); % plot 2nd row of T vs 1st row
```

It is also possible to do more sophisticated types of plots in MATLAB, including polar coordinates, three-dimensional surfaces, contour plots, ...

Example 1. Plot the triangle with vertices $(-0.5,-1)$, $(0,1)$, $(0.5,-1)$

```
x = [-0.5, 0, 0.5]; % x data
y = [-1, 1, -1]; % y data
T = [x;y]; % 2x3 matrix with x,y coordinates
T = [T, T(:,1)]; % repeat 1st point to close triangle
t = plot(T(1,:),T(2,:),'linewidth',3); % draw triangle
                  % t is handle to graphics object
axis equal; % guarantees 1-1 aspect ratio
axis([-2,2,-2,2]); % visualization window
grid on; % add reference grid
print -depsc triangle.eps; % or -djpeg90 triangle.jpg
```



2D rotations, reflections, scaling

- The matrix of a rotation of angle θ in the plane is

$$A = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

- The matrix of a reflection in the plane w.r.t. line $y = x\tan\frac{\theta}{2}$ is

$$A = \begin{bmatrix} \cos\theta & \sin\theta \\ \sin\theta & -\cos\theta \end{bmatrix}$$

- The matrix of a scaling in the plane by a factor s_x in the x-direction and s_y in the y-direction is

$$A = \begin{bmatrix} s_x & \\ & s_y \end{bmatrix}$$

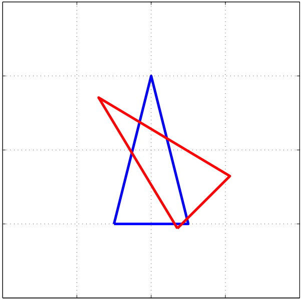
- To apply A to $T = \begin{bmatrix} x_1 & x_2 & \cdots & x_N \\ y_1 & y_2 & \cdots & y_N \end{bmatrix}$ simply multiply A and T

```
AT = A*T;
```

The matrix AT contains the coordinates of the transformed points

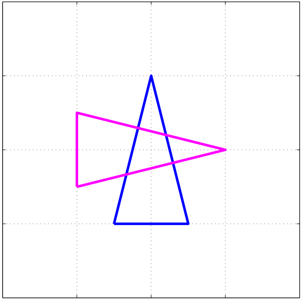
Example 2. **Rotate the triangle from Example 1 counterclockwise by** 45^o

```
hold on; % superpose on current plot
ang = pi/4; % rotation angle of 45 degrees counterclockwise
A = [cos(ang), -sin(ang); sin(ang), cos(ang)]; % rotation matrix
AT = A*T; % apply rotation to triangle
t2 = plot(AT(1,:),AT(2,:),'r','linewidth',3);
        % rotated triangle in red
        % t2 is a handle to the graph of the rotated triangle
hold off; % release plot
```



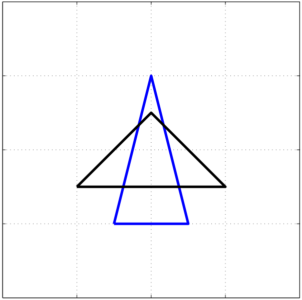
Example 3. Reflect the triangle from Example 1 along the line $y = x$

```
delete t2; % delete rotated triangle from Example 2
hold on;
ang = pi/2; % tan(ang/2)=1=slope of line => ang/2=pi/4
A = [cos(ang), sin(ang); sin(ang), -cos(ang)]; % reflection
AT = A*T; % apply reflection to triangle
t3 = plot(AT(1,:),AT(2,:),'m','linewidth',3); % in magenta
hold off;
```



Example 4. **Scale the triangle from Example 1 by $s_x = 2$ and $s_y = .5$**

```
delete t3;
hold on;
sx = 2; sy = .5; % scaling factors in x and y directions
A = [sx, 0; 0, sy]; % scaling matrix
AT = A*T; % apply scaling to triangle
t4 = plot(AT(1,:),AT(2,:),'k','linewidth',3); % scaled in black
hold off;
```



Combinations of Transformations

- Transformations can be combined by successively applying each one

$$T \xrightarrow{A_1} A_1 T \xrightarrow{A_2} A_2 A_1 T \rightarrow \cdots$$

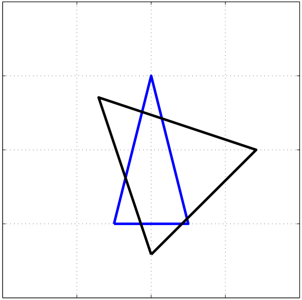
- The resulting transformation is represented by

$$A = \cdots A_2 A_1$$

Note that the first transformation appears on the right of the product of the matrix representations of the individual transformations

Example 5. Scale the triangle from Example 1 horizontally by a factor 2, then rotate it by an angle $\theta = 45^o$.

```
delete t4;
hold on;
sx = 2; sy = 1; A1 = [sx, 0; 0, sy]; % scaling
ang = pi/4; A2 = [cos(ang), -sin(ang); sin(ang), cos(ang)]; % rot
AT = A2*A1*T; % apply A1 then A2 to triangle
t5 = plot(AT(1,:),AT(2,:),'k','linewidth',3);
hold off;
```



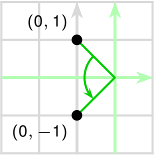
Homogeneous coordinates

- What about rotations or scalings about a point other than the origin?
- What about reflections about lines not passing through the origin?
- What about translations?
- Homogeneous coordinates $\begin{bmatrix} x \\ y \end{bmatrix} \xrightarrow{\text{extend}} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$
- The matrix of a standard transformations in homogeneous coordinates is

Rotation $\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$, **Reflection** $\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ \sin\theta & -\cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$, **Scaling** $\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$, **Translation** $\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$

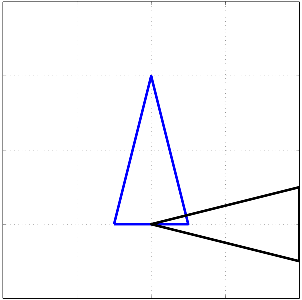
For example, to implement a rotation of angle $\frac{\pi}{2}$ about $(1,0)$:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \xrightarrow[=T(1,0)^{-1}]{T(-1,0)} \begin{bmatrix} x-1 \\ y \\ 1 \end{bmatrix} \xrightarrow{\text{Rot}(\frac{\pi}{2})} \begin{bmatrix} -y \\ x-1 \\ 1 \end{bmatrix} \xrightarrow{T(1,0)} \begin{bmatrix} 1-y \\ x-1 \\ 1 \end{bmatrix}$$



Example 6. Rotate the triangle from Example 1 counterclockwise by an angle $\theta = 90^o$ about $(1,0)$.

```
delete t5;
hold on;
tx = 1; ty = 0;
A1 = [1,0,tx;0,1,ty;0,0,1]; % translation
ang = pi/2;
A2 = [cos(ang),-sin(ang),0;sin(ang),cos(ang),0;0,0,1]; % rotation
AT = A1*(A2*(A1\T)); % (1,0)->(0,0), rotate, (0,0)->(1,0)
t6 = plot(AT(1,:),AT(2,:),'k','linewidth',3);
hold off;
```



Animation/AVI movie

```
avi = VideoWriter('triangle.avi'); avi.Quality = 50; open(avi);
writeVideo(avi,getframe);
N = 40; % number of frames for each transformation
tx = 1/N; ty = 0;
A1 = [1, 0, tx; 0, 1, ty; 0, 0, 1]; % small translation
ang = (pi/2)/N;
A2 = [cos(ang), -sin(ang), 0; sin(ang), cos(ang), 0; 0, 0, 1];
                                       % small rotation
for i = 1:N
    T = A1\T; % (1,0)->(0,0) in N steps
    set(t,'xdata',T(1,:),'ydata',T(2,:));
    writeVideo(avi,getframe); pause(0.1);
end
for i = 1:N
    T = A2*T; % rotate 90 degrees in N steps
    set(t,'xdata',T(1,:),'ydata',T(2,:));
    writeVideo(avi,getframe); pause(0.1);
end
for i = 1:N
    T = A1*T; % (0,0)->(1,0) in N steps
    set(t,'xdata',T(1,:),'ydata',T(2,:));
    writeVideo(avi,getframe); pause(0.1);
end

close(avi);
```

Instructions for the problems

For each of the following problems:

1. Create an m-file to store the MATLAB commands
2. Copy and paste the m-file into a text document
3. For Problems 1, 2, 3, include in the text document the pictures produced by MATLAB. Resize and crop the pictures so that they do not take up too much space
4. If a question requires written answers, include them in the text file in the appropriate location
5. Make sure you clearly label and separate all the questions
6. For Problem 4 you do not need to include any picture. Instead include the submission data and time of your AVI movie

Problem 1

Let $f(x) = x^2 + \sin\left(\frac{1}{x}\right)$

a) Plot f for $0 < x \leq 1$. Use an appropriate window in both x and y directions

b) Superpose a plot of the derivative of f

Let $g(x) = \frac{f(x+h) - f(x)}{h}$ with $h = 10^{-8}$.

c) Plot g for $0 < x \leq 1$. Compare to b)

d) What happens if instead $h = 10^{-15}$?

Problem 2

Consider the original triangle T. Perform the following transformations:

a) Rotate T by an angle of 180^o

b) Reflect T about the line $y = x\sqrt{3}$

c) Compose the transformations a) then b)

d) Compose the transformations b) then a). Compare to c)

e) Translate T by one unit in the positive x direction, then rotate T by an angle of 90^o about $(1, 0)$

f) Determine and implement a transformation which brings back T to its original position from the result of e)

Problem 3

Consider the square S with vertices at $(0,0)$, $(1,0)$, $(1,1)$, and $(0,1)$. Perform the following transformations:

a) Rotate S counterclockwise by an angle of 30^o about $(0,0)$

b) Translate S horizontally by one unit to the right

c) Rotate S counterclockwise by an angle of 60^o about $(1,0)$

d) Compose the transformations a) then b) then c)

e) Identify the transformation from d) and implement it directly with a single matrix

Problem 4

Create an AVI movie PB4$-$your name.avi implementing the transformations from Problem 3d) using $N = 40$ steps for each of the 2 rotations and the translation.

- Use a MATLAB command like `title ('my name')` to include your name as a title of the figure used in the video
- Submit the resulting AVI file on Blackboard