Matlab Experiment 7 Report: Geometric Transformation

Yikun Zhang*
SID: 14336275
Major: Mathematics and Applied Mathematics

Abstract

Geometry is one of the most beautiful branches of Mathematics, while transformations serve as the foundation of geometry. In this experiment we are supposed to scrutinize some interesting property of linear transformations, affine transformations, and projective transformations. More precisely, the eigenvalues and eigenvectors of linear transformations will be analyzed. Then we will jump out of the restriction of Euclidean geometry and inspect some basic concepts of Lobachevsky geometry. Last but not least, we will utilize geometric transformations to geometrically prove the correctness of the famous Fundamental Theorem of Algebra.

Contents

*School of Mathematics, Sun Yat-sen University

1 Introduction and Purpose

A geometric transformation is a function whose domain and range are sets of points.[1] The most common geometric transformations are operated within a plane. Specifically, for each point P in the xy-plane, there exists an unique point $\varphi(P)$ in the xy-plane. Then the mapping φ is a geometric transformation and $\varphi(P)$ is the image of P.

The miscellaneous geometric transformations form the main stream of classical geometric research. The beginning of formal geometry was in ancient Greek around 300 BC. The estimable Greek mathematician called Euclid took an abstract approach to geometry in his *Elements*, one of the most influential book ever written.[2] The theory introduced by Euclid has been the standard of geometric research for more than two thousand years. The so-called Euclidean geometry is an axiomatic geometry, in which all theorems ("true statements") are derived from a small number of axioms.[3] The foundation of Euclidean geometry are five postulates for plane geometry. Euclid assumes that certain constructions can be done or certain phenomena are always true:[4]

1. Draw a straight line segment between any two points.[4]
2. Extend a straight line segment indefinitely.[4]
3. Draw a circle with given center and radius.[4]
4. That all right angles are equal to one another.[3]
5. The *parallel postulate*: "That, if a straight line falling on two straight lines make the interior angles on the same side less than two right angles, the two straight lines, if produced indefinitely, meet on that side on which are the angles less than the two right angles."[3]

After 19th century, other branches of geometry burgeoned and introduced various new and interesting theories to the realm of Mathematics. For instance, differential geometry utilized the methods from calculus to study problems involving curvature and smooth manifolds.[2] In addition, the concentrations of contemporary geometric research are not restrict to shapes or transformation. The algebraic geometry has already demonstrated its power in the proof of Fermat's Last Theorem, which is a great treasure in number theory.

In the first part of this experiment, we aim to analyze the difference of shapes under some certain geometric transformations. We launch our experiments on one of the simplest transformation on the xy-plane, rotations. In addition, we will explore the properties of eigenvalues and eigenvectors of some transformations. Some empirical methods to figure out the eigenvalues of a transformation will be described as well. After the investigation of affine transformations, we will move on to a more intricate category of transformation, projective transformations. In the second part of our experiments, a particular category of noneuclidean geometry, Lobachevsky geometry, will be introduced. More precisely, we are supposed to construct the metric unit (ruler) and standard angle (compass) in Lobachevsky geometry. In the last part, we focus on the numerical proof of the Fundamental Theorem of Algebra, where some conformal mappings are involved. All our experiment is based on computer programming on the platform of Matlab 2016a, which may approximately give us the overview of these properties. Essentially, the main theme of our experiments will follow the Chapter 7 of the book "Mathematical Experiments" written by Shangzhi Li et.al[5].

2 Methods and Results

2.1 Linear Transformations and Affine Transformations

In our linear algebra course, we know that any linear transformation can be represented by a certain matrix. Define $f : V \to W$ be a linear map (or linear transformation). Then for any vector $\vec{x} \in V$, there exists a vector $\vec{y} \in W$ and a matrix A such that $\vec{y} = f(\vec{x}) = A\vec{x}$. An affine transformation g can be viewed as a generalized version of a linear transformation f, where $\vec{y} = g(\vec{x}) = A\vec{x} + vecb$. Therefore, any affine transformation can also be represented by an augmented matrix,

$$\begin{bmatrix} \vec{y} \\ 1 \end{bmatrix} = \left[\begin{array}{ccc|c} & A & & \vec{b} \\ 0 & \cdots & 0 & 1 \end{array}\right] \begin{bmatrix} \vec{x} \\ 1 \end{bmatrix}$$

One of the simplest linear transformations is a rotation in a plane, which can be represented by the matrix

$$\begin{pmatrix} cos\,\theta & -sin\,\theta \\ sin\,\theta & cos\,\theta \end{pmatrix}$$

when the rotating angle is θ. We replicate the graph 7-1 in the textbook[5] and rotate it by the angle $\frac{\pi}{3}$. To reproduce the graph, we make use of the graph of the *cosine* function within different intervals symmetric to the origin. See Figure 1 for details.

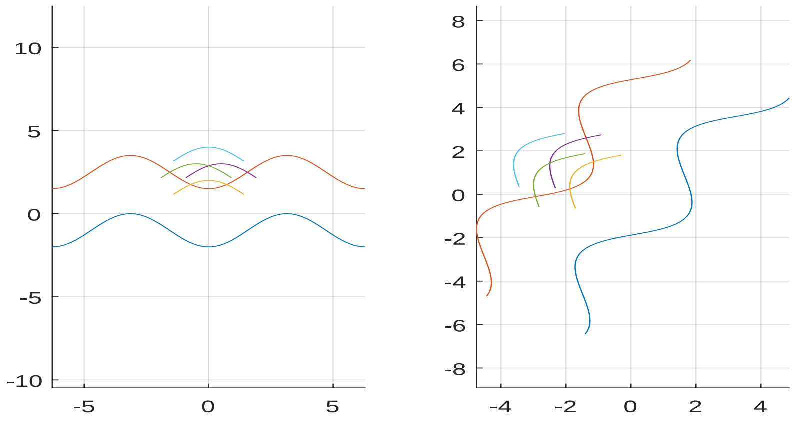


Figure 1: Comparison of Graph 7-1[5] and the graph after being rotated by $\frac{\pi}{3}$

Furthermore, we choose another matrix to transform the given graph 7-1. Here the matrix is

$$A_2 = \begin{pmatrix} 1+\epsilon_1 & \epsilon_2 \\ \epsilon_3 & 1+\epsilon_4 \end{pmatrix}$$

In the experiment, we set $\epsilon_1 = 0.01, \epsilon_2 = 0.5, \epsilon_3 = -0.01, \epsilon_4 = -0.0001$. The comparative figure goes as follows. See Figure 2 for details.

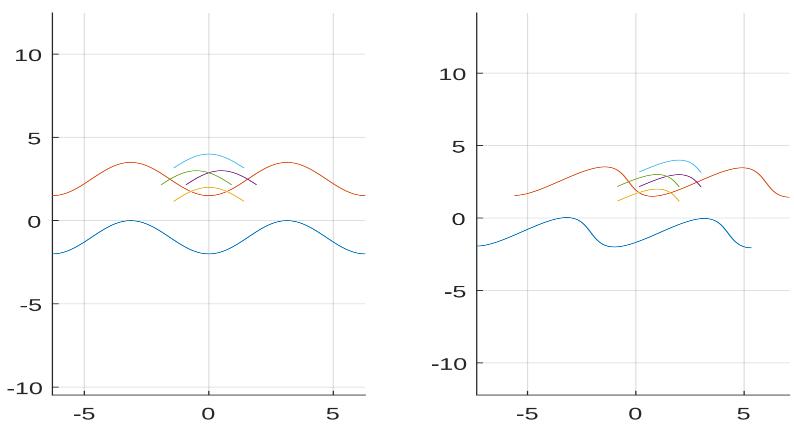


Figure 2: Comparison of Graph 7-1[5] and the graph after being transformed by A

However, one may find it difficult to inspect the deformation of a certain shape in the preceding Figure 2. Therefore, we are supposed to draw some simpler curves like the parallel straight lines or circles and see how these simple curves are transformed under A. See Figure 3 for details.

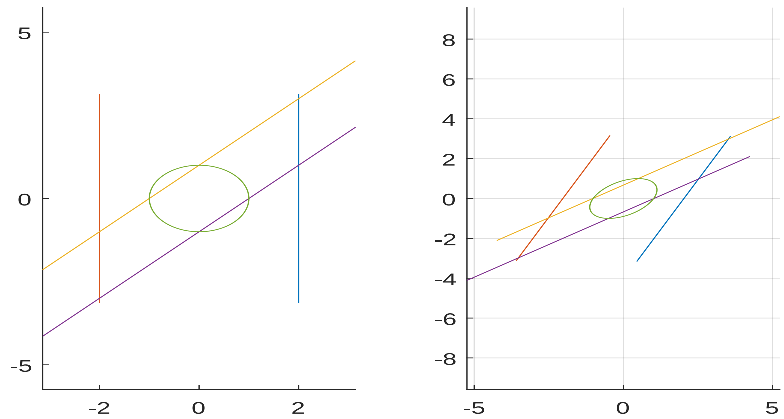


Figure 3: Comparison of some parallel straight lines and the unit circle and their shapes after being transformed by A

It turns out that linear transformations preserve parallelism and deform circles into ellipses. Careful readers may notice that the representative matrices of previous transformations are full rank, i.e., their determinants are nonzero. What the transformations would be if the representing matrix is singular? We conduct such an experiment by setting the matrix to be

$$A_3 = \begin{pmatrix} 3 & 1 \\ 6 & 2 \end{pmatrix}$$

All the curves, no matter what shapes they have, are mapped into a straight line. The reason lies in the fact that the image of such a transformation is one and thus on xy-plane it becomes a straight line. See Figure 4 for details.

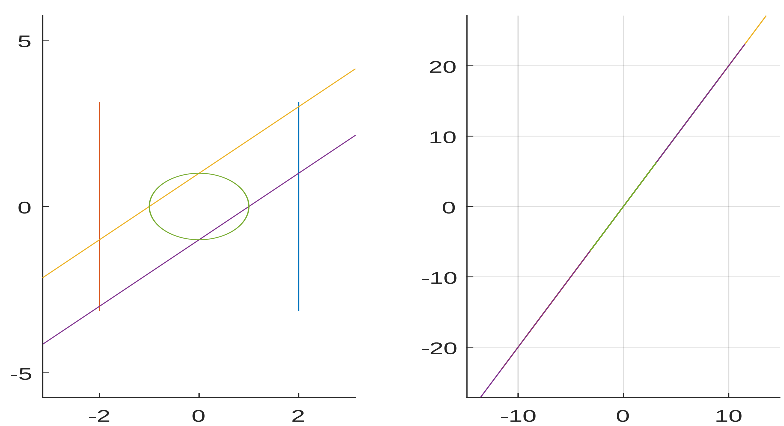


Figure 4: Transformation when A is singular

2.2 Eigenvalues of Linear Transformations

Under a (bijective) transformation, a vector will be transformed into another vector on the xy-plane. Some vectors are shifted clockwise while others counter-clockwise. However, on some particular directions, the directions of the vectors remain unchanged under the transformation, namely, $A\mathbf{u} = \lambda\mathbf{u}$. In these cases, the vector $\mathbf{u}$ is the so-called eigenvector of the transformation (or matrix) A and λ is its corresponding eigenvalue. In our experiments, the matrix is set to be

$$A = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}$$

whose eigenvalues are $\lambda_1 = 1, \lambda_2 = 3$ and the eigenvectors are $(1, -1)^T, (1, 1)^T$, respectively. Then we select n equally distributed points $P_k(cos\frac{2k\pi}{n}, sin\frac{2k\pi}{n}), k = 0, 1, ..., n-1$ on the unit circle and connect them with the origin. See Figure 5 for details.

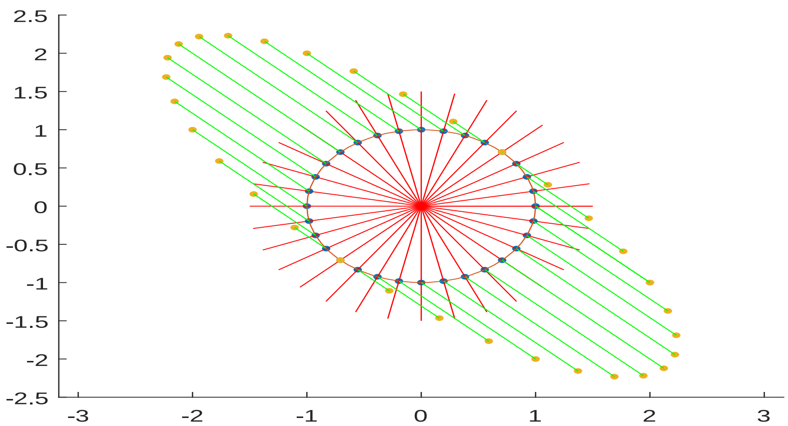


Figure 5: OP_k and its image under the transformation A

From the above figure, one can note that there are two vectors $\overrightarrow{P_kP'_k}$, whose directions coincide with the $\overrightarrow{OP_k}$. More importantly, they are on the directions of two eigenvectors. We are interested in how Graph 7-1[5] would be under the same transformation. See Figure 6 for details.

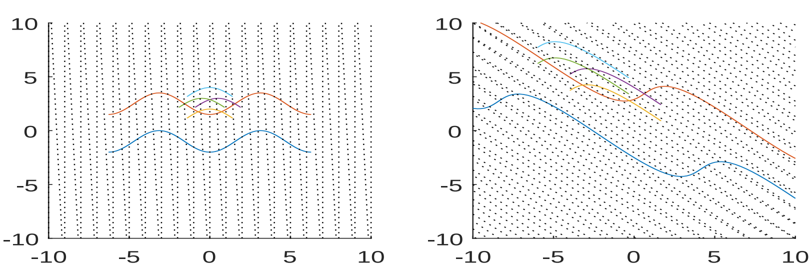
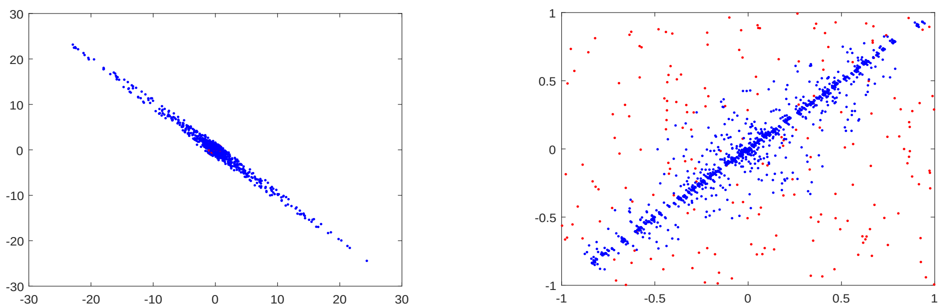


Figure 6: Graph 7-1 and grid lines under the transformation A

As is known to us, eigenvalues and eigenvectors can be obtained by solving the characteristic polynomials. However, one may wonder whether it is possible to geometrically visualize the (principle) eigenvector without analytically computing it. Here we introduced an iterative method to obtain the direction of a principle eigenvector. We simulate n points within the square $\{(x,y)||x| \leq 1, |y| \leq 1\}$ and apply the linear transformation A to them iteratively. It turns out that all the resulting points converge to the direction of the principle eigenvector as the number of iterative times increases. With the same method, we can also figure out the principle eigenvector of A^{-1}. See Figure 7 for details.



(a) Iterations under A

(b) Iterations under A^{-1}

Figure 7: Visualization of Points within a square and the resulting points after the transformation

In n-dimensional space, we can no longer visualize the direction of the principle direction. However, the method of computing eigenvector is still effective. Given a vector $\mathbf{X_0} = (x_{01}, x_{02}, ..., x_{0n})^T$, we apply the transformation iteratively to obtain an eigenvector $\mathbf{Y} = (y_1, y_2, ..., y_n)^T$. Every time before we apply the transformation, we will standardize the vector such that the summation of absolute values of its components is always 1. We set $(\frac{1}{n}, ..., \frac{1}{n})^T$ as the initial vector. See codes in the Appendix.

```
Y0=repmat(1/n,n,1);Y=Y0;
N=1000000;
for i=1:N
      Y=A4*Y;
      Y=Y./norm(Y,1);
end
norm(Y-Y0)<0.01

ans =

      1
```

2.3 Projective Transformation

Consider the following transformation $\varphi : (x, y) \to (x', y')$, where

$$x' = \frac{x}{x-1}, y' = \frac{y}{y-1}.$$

This transformation, by no means, can be considered as an affine transformation. In effect, this transformation belongs to the category of projective transformations. Under such a transformation, straight lines will be deformed into curves and concentric circles will be transformed into conic sections, namely, parabola, hyperbola, or ellipses. Essentially, the unit circle will become parabola, while circles with radii greater than 1 will become hyperbola and circles with radii less than 1 will be ellipses. See Figure 8 for details.

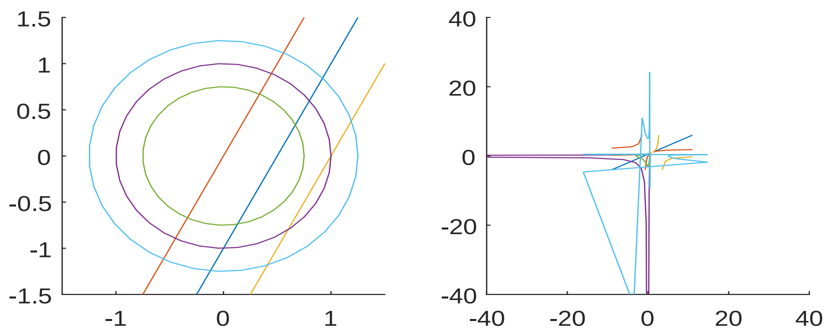


Figure 8: Projective Transformations

In general, a projective transformation is of the form,

$$x' = \frac{a_1 x + b_1 y + c_1}{a_3 x + b_3 y + c_3}, y' = \frac{a_2 x + b_2 y + c_2}{a_3 x + b_3 y + c_3}.$$

In the case when $a_3 = b_3 = 0$, we obtain an affine transformation. Hence projective transformations are generalized form of affine transformations.

2.4 Non-Euclidean Geometry

The fundamental postulate of Euclidean geometry is the parallel axiom, as stated in the introduction part. This postulate does not hold when it comes to Lobachevsky geometry, which is a typical form of non-Euclidean geometry. This violation of the Euclidean axiomatic system can be explained by the Kant model. On the Euclidean plane, all the points in the unit circle G form the whole plane of Lobachevsky geometry. If a transformation φ maps G to itself, then φ can be regarded as a Lobachevsky transformation. Besides the usual rotations centered at the origin, we give another "nontrivial" Lobachevsky transformation $\varphi : (x, y) \to (x', y')$, where

$$x' = \frac{x\, ch\alpha + sh\alpha}{x\, sh\alpha + ch\alpha}, y' = \frac{y}{x\, sh\alpha + ch\alpha}, \text{ and } ch\alpha = \frac{e^{\alpha} + e^{-\alpha}}{2}, sh\alpha = \frac{e^{\alpha} - e^{-\alpha}}{2}.$$

This is indeed a Lobachevsky transformation. See Figure 9 for details.

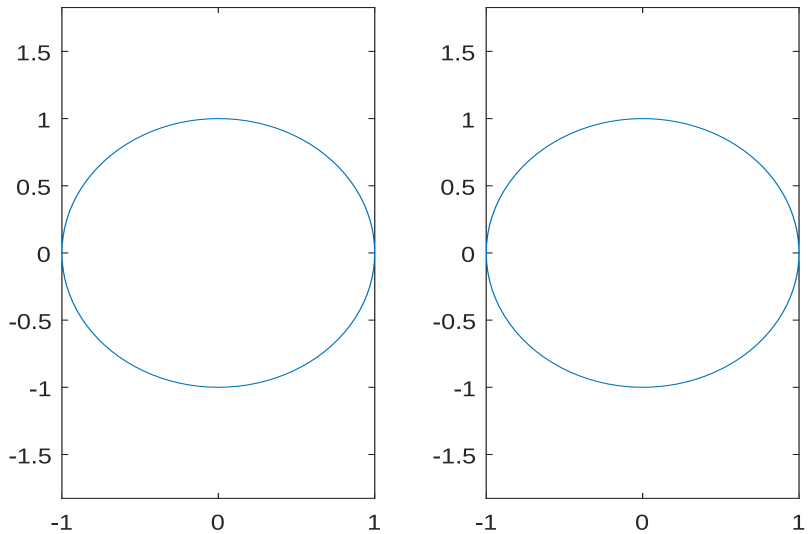


Figure 9: Verification of the Lobachevsky transformation

Next we want to construct a Lobachevsky ruler. This can be done by computing the image of the transformation on the x-axis iteratively, given the origin as the initial point, i.e., $\varphi(M_k) = \varphi(M_{k-1})$ and $M_0 = O$. As the number of iteration times increases, the image M_k gets closer to $X(1, 0)$. However, since it can never reach X, the length of OX is infinite in Lobachevsky geometry. See Figure 10 for details.

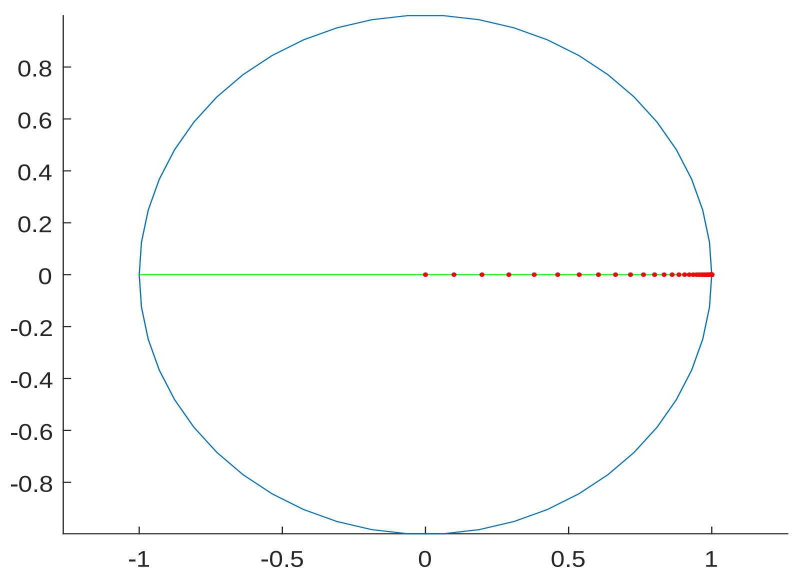


Figure 10: The Lobachevsky ruler

In order to construct Euclidean compass, we simply select n points dividing the circle into n equivalent parts. However, when we move the center to the image of the origin under the defined parabolic transformation, it will not be Euclidean compass any more. It becomes a Lobachevsky compass. See Figure 11 for details.

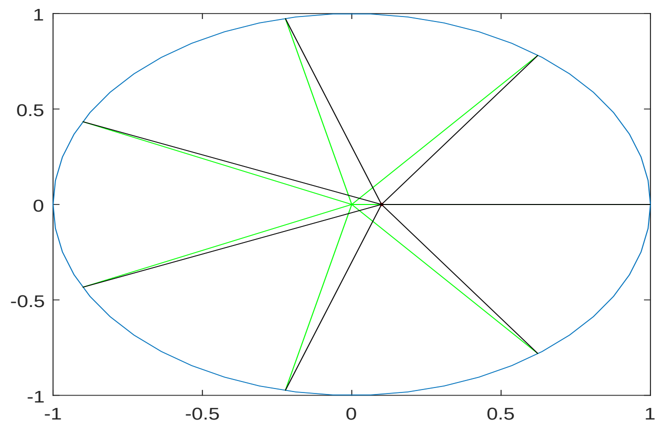


Figure 11: The Euclidean and Lobachevsky Compass

Apart from the violation of parallel axioms, there are many other counterfactual results in Lobachevsky geometry, which leads to more complicated part of geometry.

2.5 Proof of Fundamental Theorem of Algebra

Every point (x, y) on the plane can be represented by a complex number $x + yi$. Consider a conformal map $f(z) = \frac{az+b}{cz+d}$, where $a = 2, b = 1, c = 1, d = 2$ in our experiment. We are interested in how a unit circle and a straight line will become under this conformal map. See Figure 12 for details.

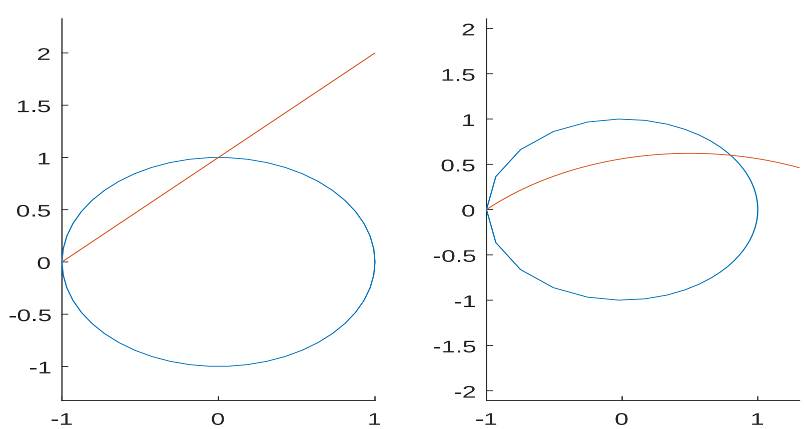


Figure 12: A Circle and Straight Line Under a conformal map

Next when the mapping becomes $f(z) = z^2$, the parallel lines along x-direction and y-direction become parabolic curves or elliptic curves. See Figure 13 for details.

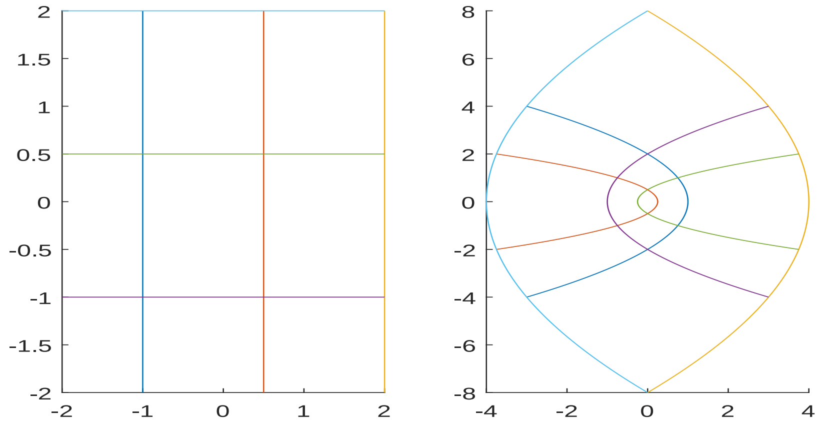


Figure 13: Parallel lines under $f(z) = z^2$

Let $C = \{x | x \in \mathbb{R}, |x| \geq 1\} \cup \{cos\theta + isin\theta | 0 \leq \theta \leq \pi\}$, which is consisted of an upper circumference of the unit circle and x-axis outside the circle. We want to identify what each part of the curve becomes under the transformation $f_1(z) = \frac{z+1}{z-1}$ and $f_2(z) = (\frac{z+1}{z-1})^2$. See Figure 14 for details.

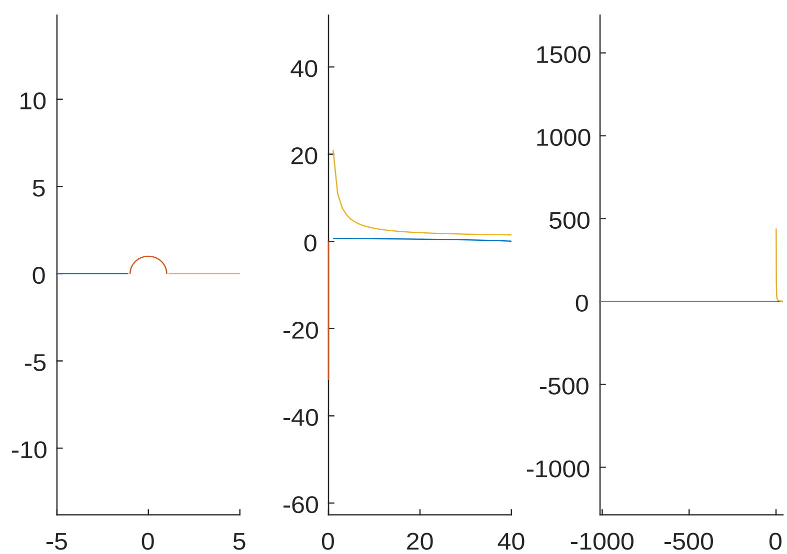


Figure 14: C under $f_1(z) = \frac{z+1}{z-1}$ and $f_2(z) = (\frac{z+1}{z-1})^2$

For example, the upper circumference of the unit circle $C_2 = \{cos\theta + isin\theta | 0 \leq \theta \leq \pi\}$ becomes the negative image axis under f_1, while the other parts are located on the real axis.

Theorem (Fundamental Theorem of Algebra). Every non-constant single-variable polynomial with complex coefficients has at least one complex root.[6]

This theorem was proved by Gauss. Up to now, there are more 100 proofs for this theorem. In complex analysis, Liouville's theorem, which states that any bounded entire function on $\mathbb{C}$ is constant. With Matlab, we are going to numerically prove the Fundamental Theorem of Algebra. Let $f(z) = z^5 + 6z^2 + 5z + 7$ and $f_1(z) = z^5$. Given circles centered at the origin, their images under f_1 are again some circles centered at the origin. Nevertheless, their images under f could be some self-joint curves. More importantly, with an appropriate choice of the radius, the curve can pass through the origin, which means that the polynomial has at least one root. Also, when the radius r is large enough, $f_1(C_r)$ asymptotically approximates $f(C_r)$. See Figure 15 for details.

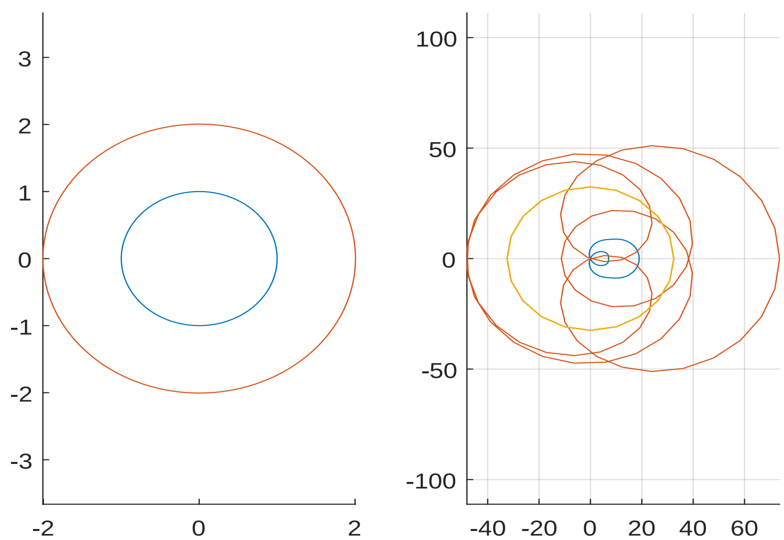


Figure 15: The images of f and f_1

When the radius r decreases from $R = 1.5$ to 0 the curve will eventually collapse to a point (the constant coefficient of the polynomial). During this process, the images will absolutely pass through at one state. The correctness of the Fundamental Theorem of Algebra is thus proved. See Figure 16 for details.

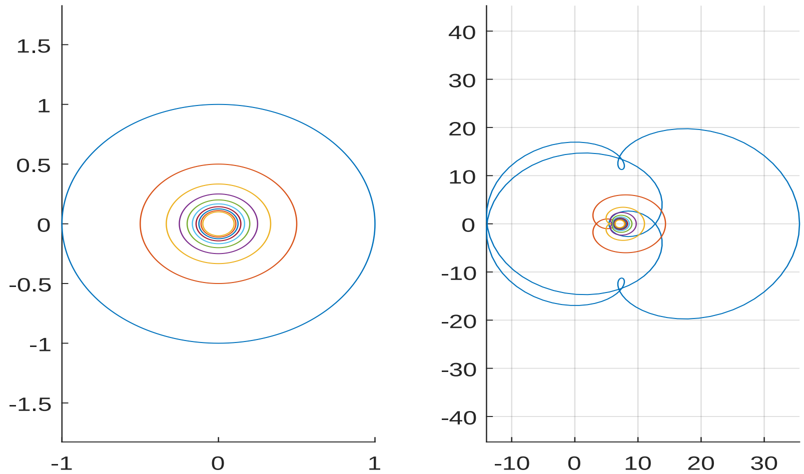


Figure 16: The deformation of images of concentric circles under f

Because of the preceding method, we can use a bisection algorithm to search for a module of a root. Basically, we choose an interval of radii such that the images of circles given radii from the interval have possibility to contain the origin or not. Then we bisect the interval until the length of the interval is smaller than a given tolerance. The end point of the interval can be regarded as the module of a root for the given polynomial. When f is defined above, the algorithm will settle down on 1.7568. See codes in the Appendix. Meanwhile, by drawing the graph of $x(\theta), y(\theta)$ in the same plotting, where $x(\theta) + y(\theta)i = f(r_0(cos\theta + isin\theta))$, we can obtain the angle of the root by computing the x-coordinate of the intersection of these two curves. See Figure 17 for details.

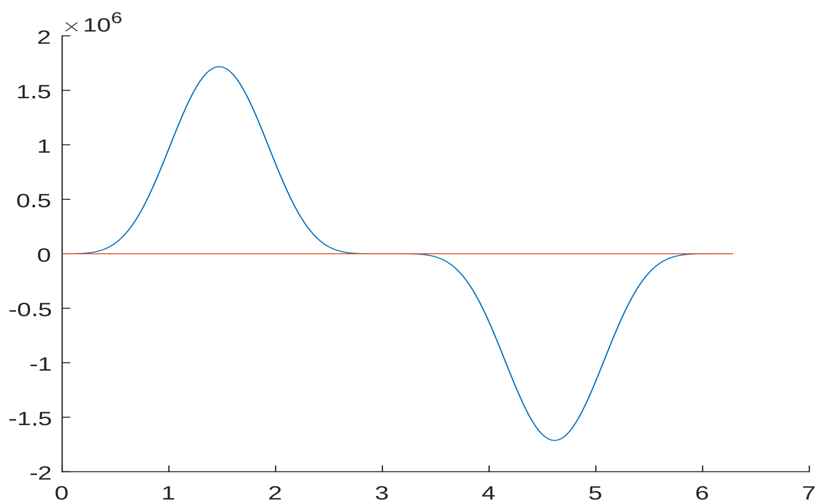


Figure 17: Plotting $x(\theta), y(\theta)$ to find out the angle of the root

Consequently, a root for $f = 0$ is $1.7568e^{3.14159296379563225343946172004 88i}$.

3 Conclusion and Discussion

In this experiment we superficially inspect some main results of geometric transformations. We also propose a numerical way to calculate the eigenvector of any matrix in any dimensional space. More importantly, our experiment follows the stream of generalizing the definition of transformations. Finally, we demonstrate that geometric transformations can also be used to prove the well-known Fundamental Theorem of Algebra.

4 Reference

[1] Wikipedia ***Geometric transformation.*** URL: `https://en.wikipedia.org/wiki/Geometric_transformation` Retrieved 26 December, 2017.

[2] Wikipedia ***Geometry.*** URL: `https://en.wikipedia.org/wiki/Geometry` Retrieved 26 December, 2017.

[3] Wikipedia ***Euclidean geometry.*** URL: `https://en.wikipedia.org/wiki/Euclidean_geometry` Retrieved 26 December, 2017.

[4] John Stillwell ***The Four Pillars of Geometry.*** Undergraduate Texts in Mathematics, 2005 Springer Science+Business Media, Inc.

[5] Shangzhi Li, Falai Chen, Yaohua Wu, and Yunhua Zhang (1999) ***Mathematical Experiments.*** Textbook Seris for 21st Century, Higher Education Press.

[6] Wikipedia ***Fundamental theorem of algebra.*** URL: `https://en.wikipedia.org/wiki/Fundamental_theorem_of_algebra` Retrieved 26 December, 2017.

5 Appendix

```
% Exercise 1 (1)
t=-2*pi:pi/20:2*pi;
p1=-cos(t)-1;p2=-cos(t)+2.5;
p3=cos(t(t<pi/2 & t>-pi/2))+1;p4=cos(t(t<pi/2 & t>-pi/2))+2;
h = figure;
subplot(1,2,1)
hold on
plot(t,p1);
plot(t,p2);
plot(t(t<pi/2 & t>-pi/2),p3);
plot(t(t<pi/2 & t>-pi/2)+0.5,p4)
plot(t(t<pi/2 & t>-pi/2)-0.5,p4)
plot(t(t<pi/2 & t>-pi/2),p4+1)
grid on
axis equal
hold off

a1=pi/3;
A=[cos(a1) -sin(a1);sin(a1) cos(a1)];
subplot(1,2,2)
hold on
point1=[t;p1];
point1=A*point1;
plot(point1(1,:),point1(2,:));
point2=[t;p2];
point2=A*point2;
plot(point2(1,:),point2(2,:));
point3=[t(t<pi/2 & t>-pi/2);p3];
point3=A*point3;
plot(point3(1,:),point3(2,:));
```

```
point4=[t(t<pi/2 & t>-pi/2)+0.5;p4];
point4=A*point4;
plot(point4(1,:),point4(2,:));
point5=[t(t<pi/2 & t>-pi/2)-0.5;p4];
point5=A*point5;
plot(point5(1,:),point5(2,:));
point6=[t(t<pi/2 & t>-pi/2);p4+1];
point6=A*point6;
plot(point6(1,:),point6(2,:));
grid on
axis equal
hold off
set(h,'Units','Inches');
pos = get(h,'Position');
set(h,'PaperPositionMode','Auto','PaperUnits','Inches','PaperSize',[pos(3), pos(4)])
print('Ex11','-dpdf','-fillpage')

% Exercise 1 (2)
h=figure;
t=-2*pi:pi/20:2*pi;
p1=-cos(t)-1;p2=-cos(t)+2.5;
p3=cos(t(t<pi/2 & t>-pi/2))+1;p4=cos(t(t<pi/2 & t>-pi/2))+2;
subplot(1,2,1)
hold on
plot(t,p1);
plot(t,p2);
plot(t(t<pi/2 & t>-pi/2),p3);
plot(t(t<pi/2 & t>-pi/2)+0.5,p4)
plot(t(t<pi/2 & t>-pi/2)-0.5,p4)
plot(t(t<pi/2 & t>-pi/2),p4+1)
grid on
axis equal
hold off

A2=[1+0.01 0.5; -0.01 1-0.0001];
subplot(1,2,2)
hold on
point1=[t;p1];
point1=A2*point1;
plot(point1(1,:),point1(2,:));
point2=[t;p2];
point2=A2*point2;
plot(point2(1,:),point2(2,:));
point3=[t(t<pi/2 & t>-pi/2);p3];
point3=A2*point3;
plot(point3(1,:),point3(2,:));
point4=[t(t<pi/2 & t>-pi/2)+0.5;p4];
point4=A2*point4;
plot(point4(1,:),point4(2,:));
point5=[t(t<pi/2 & t>-pi/2)-0.5;p4];
point5=A2*point5;
plot(point5(1,:),point5(2,:));
point6=[t(t<pi/2 & t>-pi/2);p4+1];
point6=A2*point6;
plot(point6(1,:),point6(2,:));
grid on
axis equal
hold off
set(h,'Units','Inches');
```

```
pos = get(h,'Position');
set(h,'PaperPositionMode','Auto','PaperUnits','Inches','PaperSize',[pos(3), pos(4)])
print('Ex12','-dpdf','-fillpage')

% Exercise 1 (3)
t=-pi:pi/50:pi;
x1=repmat(2,1,length(t));
x2=repmat(-2,1,length(t));
y1=t;y2=t;
x3=t;x4=t;
y3=x3+1;y4=x4-1;
x5=cos(t);y5=sin(t);
h=figure;
subplot(1,2,1)
hold on
plot(x1,y1);
plot(x2,y2);
plot(x3,y3);
plot(x4,y4);
plot(x5,y5);
axis equal
hold off

A2=[1+0.01 0.5; -0.01 1-0.0001];
subplot(1,2,2)
hold on
point1=[x1;y1];
point1=A2*point1;
plot(point1(1,:),point1(2,:));
point2=[x2;y2];
point2=A2*point2;
plot(point2(1,:),point2(2,:));
point3=[x3;y3];
point3=A2*point3;
plot(point3(1,:),point3(2,:));
point4=[x4;y4];
point4=A2*point4;
plot(point4(1,:),point4(2,:));
point5=[x5;y5];
point5=A2*point5;
plot(point5(1,:),point5(2,:));
grid on
axis equal
hold off
set(h,'Units','Inches');
pos = get(h,'Position');
set(h,'PaperPositionMode','Auto','PaperUnits','Inches','PaperSize',[pos(3), pos(4)])
print('Ex13','-dpdf','-fillpage')

h=figure;
t=-pi:pi/50:pi;
x1=repmat(2,1,length(t));
x2=repmat(-2,1,length(t));
y1=t;y2=t;
x3=t;x4=t;
y3=x3+1;y4=x4-1;
x5=cos(t);y5=sin(t);
subplot(1,2,1)
hold on
```

```
plot(x1,y1);
plot(x2,y2);
plot(x3,y3);
plot(x4,y4);
plot(x5,y5);
axis equal
hold off

A3=[3 1; 6 2];
subplot(1,2,2)
hold on
point1=[x1;y1];
point1=A3*point1;
plot(point1(1,:),point1(2,:));
point2=[x2;y2];
point2=A3*point2;
plot(point2(1,:),point2(2,:));
point3=[x3;y3];
point3=A3*point3;
plot(point3(1,:),point3(2,:));
point4=[x4;y4];
point4=A3*point4;
plot(point4(1,:),point4(2,:));
point5=[x5;y5];
point5=A3*point5;
plot(point5(1,:),point5(2,:));
grid on
axis equal
hold off
set(h,'Units','Inches');
pos = get(h,'Position');
set(h,'PaperPositionMode','Auto','PaperUnits','Inches','PaperSize',[pos(3), pos(4)])
print('Ex14','-dpdf','-fillpage')

% Exercise 2
h=figure;
n=32;t=0:2*pi/n:2*pi;t2=0:2*pi/50:2*pi;
hold on
plot(cos(t),sin(t),'.','MarkerSize',10)
plot(cos(t2),sin(t2))
plot([zeros(1,length(t));1.5*cos(t)],[zeros(1,length(t));1.5*sin(t)],'-r')
A=[2 -1; -1 1+1];
points=A*[cos(t);sin(t)];
plot(points(1,:),points(2,:),'.','MarkerSize',10)
plot([cos(t);points(1,:)],[sin(t);points(2,:)],'-g')
axis([-2.5,2.5,-2.5,2.5])
axis equal
hold off
set(h,'Units','Inches');
pos = get(h,'Position');
set(h,'PaperPositionMode','Auto','PaperUnits','Inches','PaperSize',[pos(3), pos(4)])
print('Ex2','-dpdf','-fillpage')

% Exercise 3
h=figure;
t=-2*pi:pi/20:2*pi;
p1=-cos(t)-1;p2=-cos(t)+2.5;
p3=cos(t(t<pi/2 & t>-pi/2))+1;p4=cos(t(t<pi/2 & t>-pi/2))+2;
subplot(1,2,1)
```

```
hold on
plot(t,p1);
plot(t,p2);
plot(t(t<pi/2 & t>-pi/2),p3);
plot(t(t<pi/2 & t>-pi/2)+0.5,p4)
plot(t(t<pi/2 & t>-pi/2)-0.5,p4)
plot(t(t<pi/2 & t>-pi/2),p4+1)
[X,Y] = meshgrid(-15:1:15, -15:1:15);
X = X(:);
Y = Y(:);
plot(X,Y,'k:','MarkerSize',0.001);
axis equal
axis([-10,10,-10,10])
hold off

A2=[2 -1; -1 1+1];
subplot(1,2,2)
hold on
point1=[t;p1];
point1=A2*point1;
plot(point1(1,:),point1(2,:));
point2=[t;p2];
point2=A2*point2;
plot(point2(1,:),point2(2,:));
point3=[t(t<pi/2 & t>-pi/2);p3];
point3=A2*point3;
plot(point3(1,:),point3(2,:));
point4=[t(t<pi/2 & t>-pi/2)+0.5;p4];
point4=A2*point4;
plot(point4(1,:),point4(2,:));
point5=[t(t<pi/2 & t>-pi/2)-0.5;p4];
point5=A2*point5;
plot(point5(1,:),point5(2,:));
point6=[t(t<pi/2 & t>-pi/2);p4+1];
point6=A2*point6;
plot(point6(1,:),point6(2,:));
theta = pi/4;
rot = [cos(theta) -sin(theta); sin(theta) cos(theta)]; %// Define rotation matrix
rotate_val = rot*[X Y].'; %// Rotate each of the points
X_rotate = rotate_val(1,:); %// Separate each rotated dimension
Y_rotate = rotate_val(2,:);
plot(X_rotate, Y_rotate, 'k:','MarkerSize',0.001);
axis equal
axis([-10,10,-10,10])
hold off
set(h,'Units','Inches');
pos = get(h,'Position');
set(h,'PaperPositionMode','Auto','PaperUnits','Inches','PaperSize',[pos(3), pos(4)])
print('Ex3','-dpdf','-fillpage')

% Exercise 4 (1)
h=figure;
n=200;x=-1+2*rand(1,n);y=-1+2*rand(1,n);
P1=[x;y];
A=[2 -1; -1 1+1];
plot(P1(1,:),P1(2,:),'r.')
hold on
Pt=P1;
k=3;
```

```
for i=1:k
    Pt=A*Pt;
    plot(Pt(1,:),Pt(2,:),'b.')
end
hold off
set(h,'Units','Inches');
pos = get(h,'Position');
set(h,'PaperPositionMode','Auto','PaperUnits','Inches','PaperSize',[pos(3), pos(4)])
print('Ex411','-dpdf','-fillpage')

h=figure;
n=200;x=-1+2*rand(1,n);y=-1+2*rand(1,n);
P1=[x;y];
A=[2 -1; -1 1+1];
plot(P1(1,:),P1(2,:),'r.')
hold on
Pt=P1;
k=10;
for i=1:k
    Pt=A\Pt;
    plot(Pt(1,:),Pt(2,:),'b.')
end
hold off
set(h,'Units','Inches');
pos = get(h,'Position');
set(h,'PaperPositionMode','Auto','PaperUnits','Inches','PaperSize',[pos(3), pos(4)])
print('Ex412','-dpdf','-fillpage')

% Exercise 4 (2)
n=100;A4=100.*rand(n);
Y0=repmat(1/n,n,1);Y=Y0;
N=1000000;
for i=1:N
    Y=A4*Y;
    Y=Y./norm(Y,1);
end
norm(Y-Y0)<0.01

% Exercise 5
h=figure;
x=-5:0.1:5;
y=2*x-1;
x_p=x./(x-1);y_p=y./(y-1);
subplot(1,2,1)
hold on
plot(x,y)
plot(x,y+1)
plot(x,y-1)
hold off
subplot(1,2,2)
hold on
plot(x_p,y_p)
plot(x_p,(y+1)./y)
plot(x_p,(y-1)./(y-2))
hold off

t=0:2*pi/35:2*pi;
% t=t(t~=0 & t~=pi/2 & t~=pi & t~=3*pi/2 & t~=2*pi);
subplot(1,2,1)
```

```
hold on
plot(cos(t),sin(t))
plot(0.75*cos(t),0.75*sin(t))
plot(1.25*cos(t),1.25*sin(t))
axis equal
axis([-1.5,1.5,-1.5,1.5])
hold off
subplot(1,2,2)
hold on
plot(cos(t)./(cos(t)-1),sin(t)./(sin(t)-1))
plot(0.75*cos(t)./(0.75*cos(t)-1),0.75*sin(t)./(0.75*sin(t)-1))
plot(1.25*cos(t)./(1.25*cos(t)-1),1.25*sin(t)./(1.25*sin(t)-1))
axis equal
axis([-40,40,-40,40])
hold off
set(h,'Units','Inches');
pos = get(h,'Position');
set(h,'PaperPositionMode','Auto','PaperUnits','Inches','PaperSize',[pos(3), pos(4)])
print('Ex5','-dpdf','-fillpage')

% Exercise 6 (1)
h=figure;
al=0.1;
ch=(exp(al)+exp(-al))/2;sh=(exp(al)-exp(-al))/2;
hold on
t=0:2*pi/100:2*pi;
subplot(1,2,1)
plot(cos(t),sin(t))
axis equal
subplot(1,2,2)
plot((cos(t)*ch+sh)./(cos(t)*sh+ch),sin(t)./(cos(t)*sh+ch))
axis equal
set(h,'Units','Inches');
pos = get(h,'Position');
set(h,'PaperPositionMode','Auto','PaperUnits','Inches','PaperSize',[pos(3), pos(4)])
print('Ex61','-dpdf','-fillpage')

% Exercise 6 (2)
h=figure;
al=0.1;
ch=(exp(al)+exp(-al))/2;sh=(exp(al)-exp(-al))/2;
hold on
t=0:2*pi/50:2*pi;
plot(cos(t),sin(t))
plot([-1,1],[0,0],'-g')
plot(0,0,'r.')
x_M=0;y_M=0;k=100;
for i=1:k
    x_M=(x_M*ch+sh)/(x_M*sh+ch);
    y_M=y_M/(x_M*sh+ch);
    plot(x_M,y_M,'r.')
end
axis equal
hold off
set(h,'Units','Inches');
pos = get(h,'Position');
set(h,'PaperPositionMode','Auto','PaperUnits','Inches','PaperSize',[pos(3), pos(4)])
print('Ex62','-dpdf','-fillpage')
```

```
% Exercise 6 (3)
h=figure;
n=7;theta=0:2*pi/50:2*pi;t=0:2*pi/n:2*pi;
plot(cos(theta),sin(theta))
hold on
plot([zeros(1,length(t));cos(t)],[zeros(1,length(t));sin(t)],'-g')
x_M=(0*ch+sh)/(0*sh+ch);y_M=0;
plot(x_M,y_M,'r.')
plot([repmat(x_M,1,length(t));cos(t)],[repmat(y_M,1,length(t));sin(t)],'-k')
hold off
set(h,'Units','Inches');
pos = get(h,'Position');
set(h,'PaperPositionMode','Auto','PaperUnits','Inches','PaperSize',[pos(3), pos(4)])
print('Ex63','-dpdf','-fillpage')

% Exercise 7 (1)
h=figure;
f=@(z) (2*z+1)./(z+2);
t=0:2*pi/50:2*pi;
subplot(1,2,1)
hold on
plot(cos(t),sin(t))
x=-1:0.01:1;
plot(x,x+1)
axis equal
hold off
subplot(1,2,2)
z1=cos(t)+1i*sin(t);
z2=x+1i*(x+1);
hold on
plot(f(z1))
plot(f(z2))
axis equal
hold off
set(h,'Units','Inches');
pos = get(h,'Position');
set(h,'PaperPositionMode','Auto','PaperUnits','Inches','PaperSize',[pos(3), pos(4)])
print('Ex71','-dpdf','-fillpage')

% Exercise 7 (2)
h=figure;
f2=@(z) z.^2;
y_p=-2:0.01:2;
x1=repmat(-1,1,length(y_p));
x2=repmat(0.5,1,length(y_p));
x3=repmat(2,1,length(y_p));
x_p=-2:0.01:2;
y1=repmat(-1,1,length(x_p));
y2=repmat(0.5,1,length(x_p));
y3=repmat(2,1,length(x_p));
subplot(1,2,1)
hold on
plot(x1,y_p)
plot(x2,y_p)
plot(x3,y_p)
plot(x_p,y1)
plot(x_p,y2)
plot(x_p,y3)
hold off
```

```
subplot(1,2,2)
hold on
z1=x1+1i*y_p;
plot(f2(z1))
z2=x2+1i*y_p;
plot(f2(z2))
z3=x3+1i*y_p;
plot(f2(z3))
z4=x_p+1i*y1;
plot(f2(z4))
z5=x_p+1i*y2;
plot(f2(z5))
z6=x_p+1i*y3;
plot(f2(z6))
hold off
set(h,'Units','Inches');
pos = get(h,'Position');
set(h,'PaperPositionMode','Auto','PaperUnits','Inches','PaperSize',[pos(3), pos(4)])
print('Ex72','-dpdf','-fillpage')

% Exercise 7 (3)
h=figure;
f1=@(z) (z+1)./(z-1);f2=@(z) ((z+1)./(z-1)).^2;
x1=-5:0.1:-1.1;
y1=zeros(1,length(x1));
t=0:pi/50:pi;
x2=cos(t);y2=sin(t);
x3=1.1:0.1:5;
y3=zeros(1,length(x3));
subplot(1,3,1)
hold on
plot(x1,y1)
plot(x2,y2)
plot(x3,y3)
axis equal
hold off
subplot(1,3,2)
z1=x1+1e-20i*y1;
z2=x2+1i*y2;
z3=x3+1e-20i*y3;
hold on
plot(f1(z1))
plot(f1(z2))
plot(f1(z3))
axis equal
hold off
subplot(1,3,3)
hold on
plot(f2(z1))
plot(f2(z2))
plot(f2(z3))
axis equal
hold off
set(h,'Units','Inches');
pos = get(h,'Position');
set(h,'PaperPositionMode','Auto','PaperUnits','Inches','PaperSize',[pos(3), pos(4)])
print('Ex73','-dpdf','-fillpage')

% Exercise 8 (1)
```

```
h=figure;
t=0:pi/50:2*pi;n=5;
f=@(z) z.^n+6*z.^2+5*z+7;
f1=@(z) z.^n;
Roots=roots([1 0 0 6 5 7]);
subplot(1,2,1)
hold on
plot(cos(t),sin(t))
plot(norm(Roots(1))*cos(t),norm(Roots(1))*sin(t))
axis equal
hold off
subplot(1,2,2)
z_p=cos(t)+1i*sin(t);
z_p2=norm(Roots(1))*cos(t)+1i*norm(Roots(1))*sin(t);
hold on
plot(f(z_p))
plot(f(z_p2))
plot(f1(z_p2))
axis equal
grid on
hold off
set(h,'Units','Inches');
pos = get(h,'Position');
set(h,'PaperPositionMode','Auto','PaperUnits','Inches','PaperSize',[pos(3), pos(4)])
print('Ex81','-dpdf','-fillpage')

% Exercise 8 (2)
h=figure;
k=10;t=0:pi/100:2*pi;subplot(1,2,1)
hold on
for i=1:k
    plot(cos(t)/i,sin(t)/i)
end
axis equal
hold off
subplot(1,2,2)
hold on
for i=1:k
    z_pp=1.5*cos(t)/i+1i*1.5*sin(t)/i;
    plot(f(z_pp))
end
axis equal
grid on
hold off
set(h,'Units','Inches');
pos = get(h,'Position');
set(h,'PaperPositionMode','Auto','PaperUnits','Inches','PaperSize',[pos(3), pos(4)])
print('Ex82','-dpdf','-fillpage')

% Exercise 8 (3)
h=figure;
rl=10;rr=0;t=0:pi/100:2*pi;
while abs(rl-rr)>1e-5
    rmid=(rl+rr)/2;
    z_pm=rmid*cos(t)+1i*rmid*sin(t);
    fm=f(z_pm);
    [sortm,sortinm]=sort(abs(imag(fm)));
    first6=sortinm(1:6);
    rem=fm(first6);
```

```
    [~,index]=min(abs(real(rem)));
    rem=real(rem(index));
    if rem>0
        rr=rmid;
    else
        rl=rmid;
    end
end
rop=rr

syms a
x=real(f(rop*(cos(a)+i*sin(a))));
y=imag(f(rop*(cos(a)+i*sin(a))));
hold on
plot(t,subs(x,a,t))
plot(t,subs(y,a,t))
hold off
phi=vpasolve(x==0,a,[0,2*pi])
set(h,'Units','Inches');
pos = get(h,'Position');
set(h,'PaperPositionMode','Auto','PaperUnits','Inches','PaperSize',[pos(3), pos(4)])
print('Ex83','-dpdf','-fillpage')
```