ELEN 161
A. I. Zecevic

Project 1: Fractals, the Mandelbrot Set and Julia Sets

Problem 1. In this problem, you are asked to examine how simple recursive rules can give rise to highly complex patterns, some of which closely resemble forms that are encountered in nature.

(a) Describe the axiom and the transformation rule for each of the six recursive schemes shown in Table 1. In all cases, the first F should be interpreted as a single vertical line pointing upward.

	Axiom	Rule	Angle
Scheme 1	F - - F - - F	F + F - - F + F	60°
Scheme 2	F + F + F + F	F + F - F - FF + F + F - F	90°
Scheme 3	F + F + F + F	FF [+ F + F + F] F	90°
Scheme 4	F	F [+ F - F - F] F [- F - F - F] F	90°
Scheme 5	F	F [+ F] F [- F] F	25°
Scheme 6	F	FF - [- F + F + F] + [+ F - F - F]	25°

Table 1. Axioms and rules for Schemes 1 - 6.

(b) Use the software package described in the tutorial to plot:

(*i*) The third iteration for Schemes 1 and 2

(*ii*) The third and fifth iterations for Schemes 3, 4 and 5

(*iii*) The third and fourth iterations for Scheme 6

(c) How are the first four patterns different from the last two, and what can you conclude about biological patterns of growth in nature based on your simulation results? Explain.

Problem 2. The Mandelbrot set represents the set of complex numbers c for which the sequence

$$z(n+1) = z^2(n) + c \tag{1}$$

is *bounded* when $z(0) = 0$. The structure of this set is remarkably complex, and the resulting fractal is considered to be one of the most aesthetically pleasing objects in all of mathematics.

(a) Compute this set using function mandelbrot.m with the following parameters:

(i) Grid boundaries: $X_{\text{min}} = -2$; $X_{\text{max}} = 0.5$; $Y_{\text{min}} = -1.25$; $Y_{\text{max}} = 1.25$

(ii) Number of iterations: $K = 100$

(iii) Convergence criterion: $|z(n)| \leq 2$ for all $n \leq K$

(iv) Spacing between points on the grid: $s = 1/2,500$

Note. This choice of s will give you a $6,250 \times 6,250$ grid, with roughly 39 million points. You will actually need that level of detail to obtain a reasonably accurate representation of the Mandelbrot set. To improve the visual appearance of this plot, use the “axis square” command in Matlab.

(b) Zoom into at least two “interesting” areas of the plot obtained in part (a) by changing parameters X_{min}, X_{max}, Y_{min} and Y_{max}. Keep $K = 100$, and choose s so that you end up with a $5,000 \times 5,000$ grid (which amounts to 25 million points). The kinds of images that you might get are illustrated in Figs. 1 and 2 (which, of course, are just the “tip of the iceberg”). Feel free to explore further.

Problem 3. Filled Julia sets are obtained from the same equation as the Mandelbrot set. The difference between the two is that parameter c is fixed in this case, and what is varied is the initial condition $z(0)$. Once again, the objective is to identify a set of points for which sequence $z(n)$ remains bounded.

(a) Modify function mandelbrot.m to produce Julia sets for different choices of c.

(b) Vary parameter c in such a way that its real and imaginary parts satisfy $-1 \leq \text{Re}(c) \leq 1$ and $0 \leq \text{Im}(c) \leq 1$, respectively. Identify at least two Julia sets that you find aesthetically pleasing, and plot them.

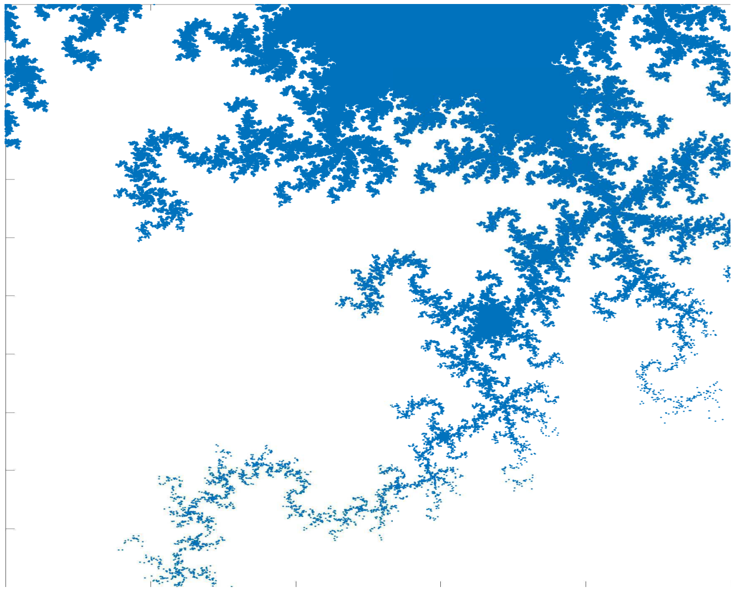

Figure 1: Mandelbrot zoom 1

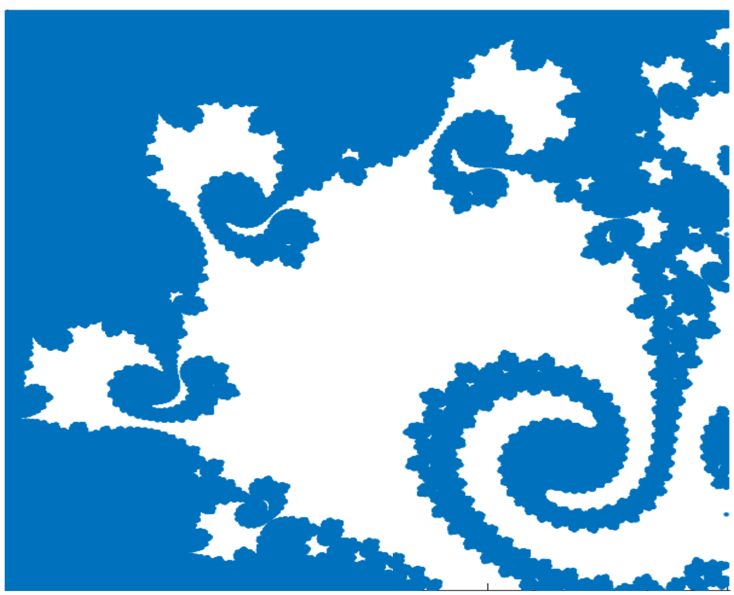

Figure 2: Mandelbrot zoom 2

Note. Make sure that you record the corresponding value of c for each plot in Problem 3(b). You can use the following parameters for your simulation:

(i) Grid boundaries: $X_{\min} = -2$; $X_{\max} = 2$; $Y_{\min} = -2$; $Y_{\max} = 2$

(ii) Number of iterations: $K = 100$

(iii) Convergence criterion: $|z(n)| \leq 2$ for all $n \leq K$

(iv) Spacing between points on the grid: $s = 1/2,000$

Figures 3, 4 and 5 illustrate the kinds of forms that you might obtain.

Problem 4. The notion of a Julia set is not limited to equation (1), and can be extended to general sequences of the form

$$z(n+1) = F[z(n)] + c \tag{2}$$

where function $F(z)$ contains higher powers of z and possibly sines and cosines as well. The procedure in this case is essentially the same as before - you need to fix a value for c and identify all initial conditions $z(0)$ for which $|z(n)| \leq R$ for all $n \leq K$. The only real difference is that R needn't be 2 any more, and depends on the choice of function F.

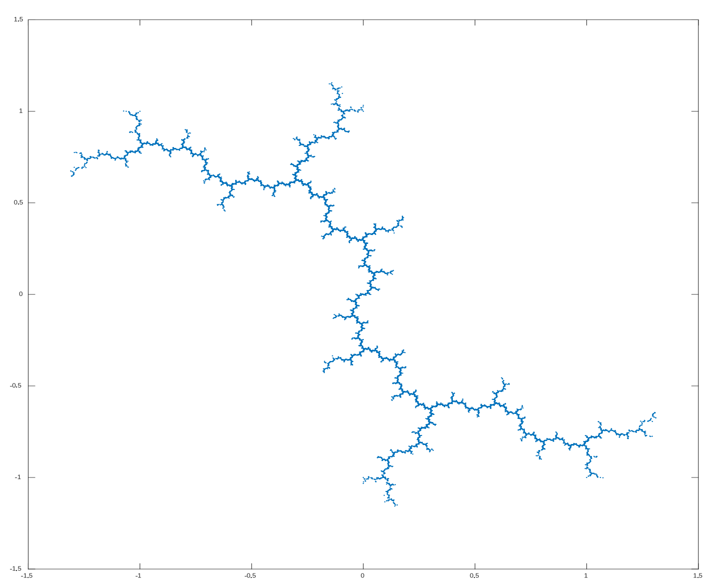

Figure 3: Julia set, variant 1

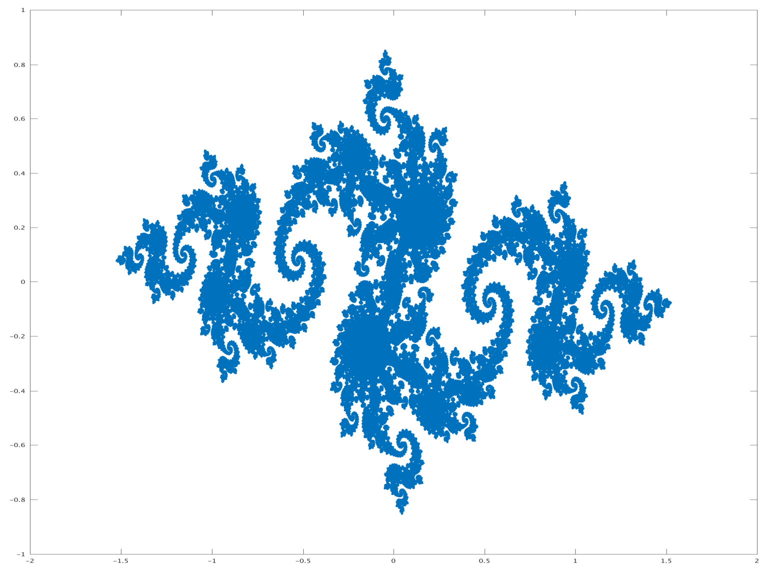

Figure 4: Julia set, variant 2

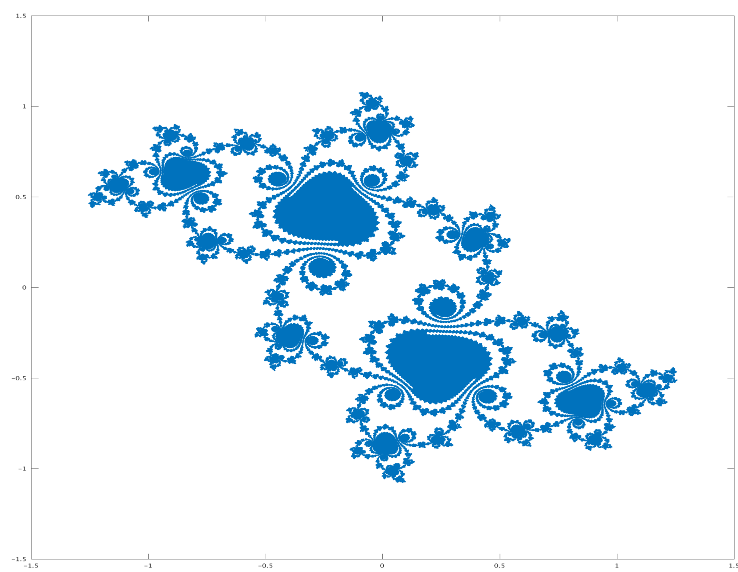

Figure 5: Julia set, variant 3

The set of points obtained in this way can produce some very interesting geometric forms if certain simple modifications are made to the convergence criterion. It is interesting to note that this property was discovered accidentally, and was the result of a programming error. Instead of assuming that system (2) diverges when $|z(n)| > R$, the altered code included a *second* (and completely unnecessary) condition, which required that *either* $|\operatorname{Re}[z(n)]| < R$ *or* $|\operatorname{Im}[z(n)]| < R$. Both criteria needed to be met before the sequence could be classified as "unbounded".

Although adding this second requirement has no mathematical justification, the resulting images are remarkably interesting. They do not represent actual Julia sets since they contain some points for which the sequence diverges, but they bear a striking resemblance to living organisms (which is completely unexpected).

(a) Modify the m-file that you developed in Problem 3 to allow for general sequences like the one in equation (2), and include the "erroneous" divergence criterion described above.

(b) Use the m-file obtained in part (a) to compute "quasi-Julia sets" for the three systems shown below:

System 1

$$z(n+1) = z^2(n) + z^6(n) + c \tag{3}$$

System 2

$$z(n+1) = z^2(n) + \sin z(n) + c \tag{4}$$

System 3

$$z(n+1) = z^3(n) + c \tag{5}$$

In all three cases, you can use the following parameters in your simulation:

(i) Grid boundaries: $X_{\min} = -2$; $X_{\max} = 2$; $Y_{\min} = -2$; $Y_{\max} = 2$

(ii) Spacing between points on the grid: $s = 1/1,000$

(iii) Convergence criterion: $|z(n)| \leq 10$ for all $n \leq K$

Note that the values for K are *not* uniform in this problem, and should be set to $K = 5$ for Systems 1 and 2, and $K = 10$ for System 3. The choice of parameter c is up to you, the only restriction being that $0 \leq c \leq 1$. Your task will be to identify values that produce aesthetically pleasing "biological" forms for each of the three cases. To give you an idea of what you should be looking for, some possible scenarios are shown in Figs 6, 7 and 8.

(c) In order to compare the sets obtained in part (b) with “regular” Julia sets, remove the unnecessary divergence condition from the m-file that you created in part (a). Plot the resulting Julia sets for the values of c that you chose in part (b), using the same parameters. What you will find is that these plots aren’t nearly as interesting as the previous ones (Figs. 9, 10 and 11 illustrate what I mean by that).

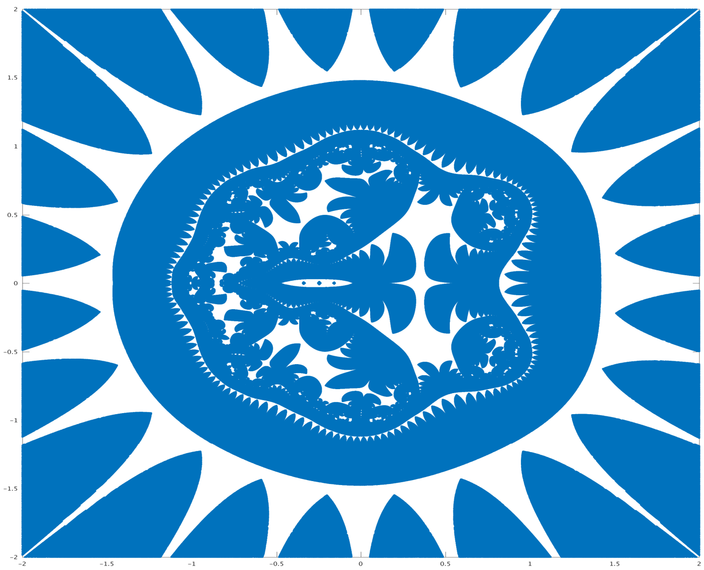

Figure 6: “Biological” form 1

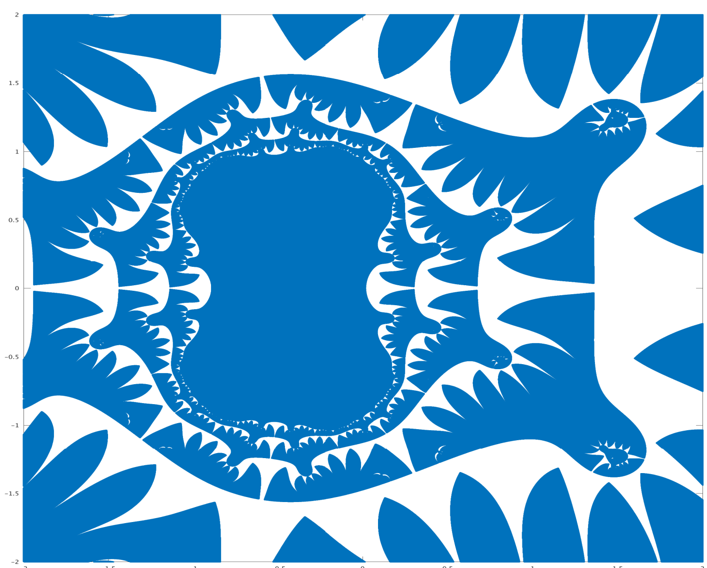

Figure 7: “Biological” form 2

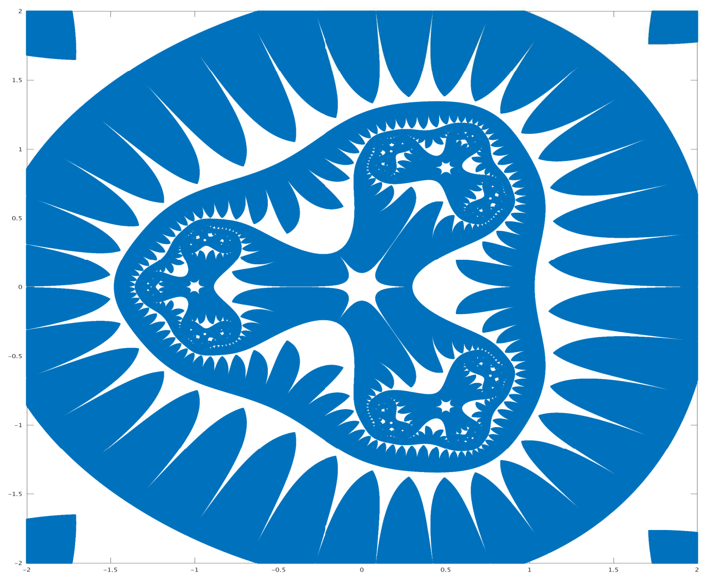

Figure 8: “Biological” form 3

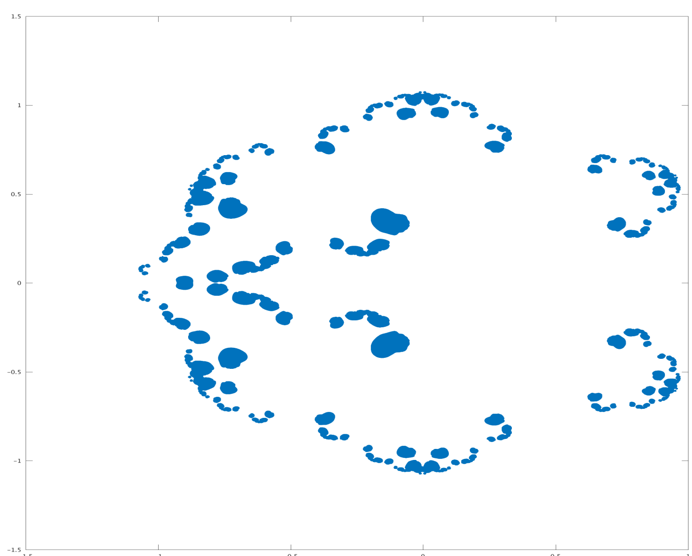

Figure 9: Julia set that corresponds to Fig. 6.

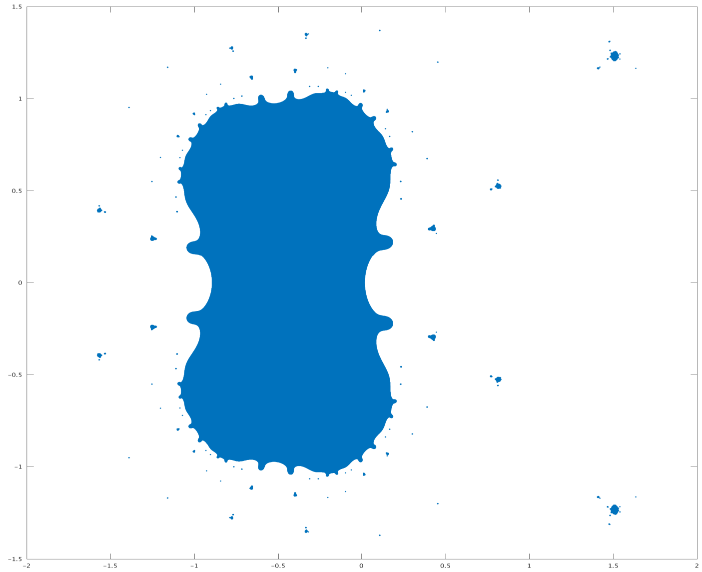

Figure 10: Julia set that corresponds to Fig. 7.

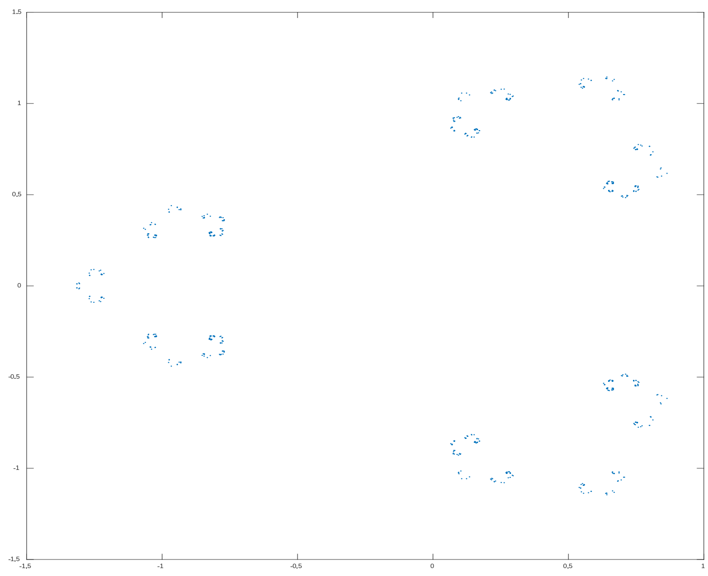

Figure 11: Julia set that corresponds to Fig. 8.