THREE DIMENSIONAL JULIA SETS

Nitin Uchil[1]

1. JULIA SETS

Consider the dynamics of the following function in the complex plane:

$$z \rightarrow F(z) + c \qquad (1.1)$$

On iterating the above function, we will have points which diverge to infinity, converge to a fixed point, oscillate between points, or keep having different values on continued iterations. *For a given c, the Julia set is defined as the set of all complex z for which the map iteration remains bounded forever.* The Julia set of fractals were first conceived by G. Julia and P. Fatou around 1918. However, this field remained dormant until B. Mandelbrot revealed the striking beauty and intricacy of these shapes in the complex plane. Mathematicians Douady, Hubbard, Milnor and Pietgen have since explored this sets intricacies, developing and proving mathematical conjectures in the course of their computer aided explorations.

Figure 1 shows the Julia set for the Glynn mapping given by:

$$F(z) = z^{1.5} - 0.2 \qquad (1.2)$$

The dynamics of the quadratic map:

$$z \rightarrow z^2 + c \qquad (1.3)$$

gives points on the complex plane which either diverge to infinity, converge or remain bounded. Figure 2 shows Julia sets for this quadratic maps for various values of c along the real axis. Figure 3 shows Julia sets for various values of c along the imaginary axis. It is seen that the complex plane of initial values is subdivided into two subsets. The first one collects points for which the iteration escapes: this is called the escapee set E. The iteration of all other initial values remains in a bounded region forever, and we collect these points in the so-called prisoner set P or basin of attraction. The boundary between E and P is called the Julia set of the iteration. Some of the salient features of Julia sets in two dimensions for the quadratic iterator are:

1. The Julia set is invariant under iteration. This feature is used to find the points on the Julia set for a particular function using the pre-image technique or the Inverse Iteration Mehod.
2. Julia sets are either connected (there is only one prisoner set) or Cantor dust.This is what is known as the structural dichotomy of Julia sets. The Julia set is one piece (connected) if and only if the iteration of the critical point is bounded. The Julia set is a Cantor set if and only if the iteration of the critical point leads to infinity.
3. The prisoner set P is topologically equivalent to a circle (that is, it is a simple curve).
4. The Mandelbrot set $M = \{ c \in Z \mid J_c \text{ is connected} \}$ gives an ordering scheme for the Julia sets.

The dynamics of the cubic map:

$$z \rightarrow z^3 + c \qquad (1.4)$$

is depicted in Figures 4 and 5.

2. QUATERNIONS

2.1 Definition

The Irish physicist and mathematician Sir William Rowan Hamilton invented the quaternion which can be thought of as a representation of numbers in four space. *Hamilton called his vectors triplets, because forces act in three dimensions, and in course of time he was anxious to find a way for their multiplication. His home circle became interested in this puzzle and every morning, on coming down to breakfast, one of his little boys used to ask, 'Well, Papa, can you multiply triplets? Whereto he was obliged to answer with a sad shake of his head, 'No, I can only add and subtract them.' But one day, so he relates in his usual Irish exuberance, he was walking with his wife beside the Royal canal on his way to a meeting in Dublin of the Academy. Although she talked with him now and then, yet an undercurrent of thought was going on in his mind, which at last gave result. It came in a very tangible form, and it at once suggested to him many a long year of purposeful work upon an important theme. He could not resist the impulse to cut with a knife, on the stone of Brougham Bridge, as they passed it, the fundamental formulae:*

$$i^2 = j^2 = k^2 = ijk = -1 \qquad (2.1)$$

indicative of the quaternions which gave the solution of the problem (7).

Quaternions have since been used to describe the dynamics of motion in 3-space. The Space Shuttle's flight software uses quaternions in its computations for guidance, navigation, and flight control for reasons of compactness, speed and avoidance of singularities (11).

Typically, a quaternion q E H can be represented by the symbol:

$$q = a + ib + jc + kd \qquad (2.2)$$

where $a + ib$ is the regular definition of a complex number, and j and k denote two additional imaginary units with components c and d, respectively. Almost all rules for real and complex numbers holds in quaternion space. The only exception is multiplication which is non commutative and can be defined by the following:

$$\begin{array}{ccccc} ij & = & -ji & = & k \\ jk & = & kj & = & i \\ ki & = & -ik & = & j \end{array} \qquad (2.3)$$

Thus the Multiplicative Rule (Rule X) can be represented by the following:

X	1	I	j	k
1	1	I	j	k
I	I	-1	k	-j
j	j	-k	-1	i
k	k	j	-I	-1

(2.4)

Consider the multiplication of two quaternions given by

$$q = q_1 q_2 \qquad (2.5)$$

where:

$$q_1 = a_1 + ib_1 + jc_1 + kd_1$$

and

$$q_2 = a_2 + ib_2 + jc_2 + kd_2$$

Thus we can represent q_1q_2 by:

$$q = q_1q_2 = a + ib + jc + kd \quad (2.6)$$

where :

$$a = a_1a_2 - b_1b_2 - c_1c_2 - d_1d_2$$
$$b = a_1b_2 + b_1a_2 + c_1d_2 - d_1c_2$$
$$c = a_1c_2 - b_1d_2 + c_1a_2 - d_1b_2$$
$$d = a_1d_2 + b_1c_2 - c_1b_2 + d_1a_2$$

3. QUATERNION JULIA SETS.

The dynamics of quadratic functions have been observed mainly in the complex plane. However as shown by Alan Norton (9), they exist in the 4-D space of the quaternions as well. By using the rules of quaternion algebra, the Julia set for the following iterate can be computed:

$$q \rightarrow F(q) + q_0 \quad (3.1)$$

In this project, the quadratic iterator is studied:

$$q \rightarrow q^2 + q_0 \quad (3.2)$$

Iterating the above for points in 4-space , the dymanics of the function (what is typically referred to as the explosion of the funcion) is obtained. Since the complex plane is a subset of the quaternions, the same complex Julia sets exist in the quaternions but often have extensions outside the complex plane. An interesting property about quaternion Julia sets is that given two complex Julia sets differing only by a rotation about the origin, their supersets in 3-D may have completely different shapes.

4. RENDERING TECHNIQUES

Quaternion representations are so complicated that it is useful to develop methodologies to aid in their display. Such methods reveal an exotic visual universe of forms. Alan Norton is the pioneer in 3D representations of quaternion iteration, and he has displayed the surface texture of 3D slices of 4D quaternions. Quaternion Julia sers can be visualized in 3-D by examining the intersection of the 4-space with a three space such as that spanned by 1, i, j, at 0k. There are various techniques developed for the definition of the boundary of the Julia set and some of them are:

1. Pixel mapping.
2. Boundary tracing or Pre-imaging or Inverse iteration
3. Ray tracing.

4.1 PIXEL MAPPING

Pixel mapping is done in two methods:

1. Interactive
2. Batch

In interactive processing for two dimensional objects, a random point is generated within the confines of the frame and the pixel value of this point is got by iterating it wrt the characteristic function. Thus, if you have a really slow computer or a large number of points to iterate you see the image "grow" in front of your eyes.

In batch processing, typically like the ones I have developed, a mesh is developed for the frame extents in consideration and each grid point undergoes the "hit-test" to get its contouring value which are stored in post-processing registers to be later rendered by a polygon sweep method where the intermediate values are obtained by contour interpolation. The advantages of this technique is that it gives a uniform plot but the values between two points are only the interpolated colors and do not represent the true behavior of the system. Thus, finer the mesh, better the result.

In mixed mode processing, batch processing is done in parts and these are intermittently rendered. Thus the amount of memory required to save the post processing data is reduced. In some techniques, a sort of a piping schedule is employed where the length of the pipe is the memory available and plotting starts as soon as this length is filled.

4.2 BOUNDARY TRACING

4.3 RAY TRACING

Ray-traced images are created by numerically simulating the behavior of photons impinging on a mathematical film plane, through the principles of geometric optics. The ray tracing approach has been one of the most successful techniques to date in creating high resolution computer generated images, and is becoming progressively more practical due to the increasing speed and decreasing cost of computation. Julia sets in the quaternions can be viewed using high resolution computer graphics as objects modeled from clay and illuminated from the outside.

5. RENDERING THREE DIMENSIONAL JULIA SETS

6. APPENDIX

6.1 MANDELBROT SETS FOR HIGHER ORDER EQUATIONS

Figure 6 shows Mandelbrot set for orders ranging from 2 to 10. It is seen that a Mandelbrot set of order n is n-1 symmetric in the Argand plane and following is a proof by induction why this is so:

Proof: A Mandelbrot set of order n is defined by:

$$z \rightarrow z^n \text{ at } z_0$$

For a z that converges we thus have to prove that:

$$f(z) = z^n \text{ is n-1 symmetric.}$$

1. If n=1, clearly it is non-symmetric, ie there are 1-1=0 axes of symmetry.

2. For n=2, $f(z)=z^2$, tpt: $f(z) = f(z+\pi)$

$$
\begin{aligned}
f(z) &= Re^{2i\theta} \\
f(z+\pi) &= Re^{2i(\theta+\pi)} \\
&= e^{2i\pi}Re^{2i\theta} \\
&= Re^{2i\theta} \\
&= f(z)
\end{aligned}
$$

3. For n=r, $f(z)=z^r$, tpt: $f(z) = f(z+2\pi/r)$

$$
\begin{aligned}
f(z) &= Re^{ri\theta} \\
f(z+2\pi/r) &= Re^{ri(\theta+2\pi/r)} \\
&= e^{2i\pi}Re^{ri\theta} \\
&= Re^{ri\theta} \\
&= f(z)
\end{aligned}
$$

(QED)

Figure 7 shows Julia Sets for orders ranging from 2 to 10.

6.2 ROTATION OF JULIA SETS

The rotation of a Julia set in the complex plane (5) is computed by incorporating the homemorphism:

$$g(z) = e^{i\theta}z \quad (6,1)$$

into the iterated function

$$h(z) = g \circ f \circ g^{-1}(z) \quad (6.2)$$

Consider a polynomial Julia set of order n given by:

$$z \rightarrow z^n + c \quad (6.3)$$

We have:

$$g^{-1}(z) = z/e^{i\theta} \quad (6.4)$$

Thus:

$$f \circ g^{-1}(z) = z^n/e^{ni\theta} + c \quad (6.5)$$

And:

$$g \circ f \circ g^{-1}(z) = -e^{(n-1)i\theta}z^n + e^{i\theta}c \quad (6.6)$$

7. REFERENCES

1. *Barnsley, M. J.,* "Fractals Everywhere", Academic Press.

2. *Brand, Louis,* "Vector and Tensor Analysis", John Wiley & Sons pp 403-429.

3. *Briggs, John and Peat, F. David* "Turbulent Mirror" Harper and Row.

4. *Foley, J. D., van Dam, A.., Feiner, S. K., Hughes, J. F.,* "Computer Graphics: Principles and Practice", Addison-Wesley.

5. *Hart, J.C., Sandin, D.J., Kaufmann, L.H.,* "Ray tracing deterministic 3-D fractals", Computer Graphics 23, 3 (1989) pp. 91-100.

6. *Mandelbrot, B. B.* "The Fractal Geometry of Nature", W.H. Freeman and Co.,

7. *Newman , James R.* "The World of Mathematics - Volume One ", Tempus Publications

8. *Norton, V. A.,* "Generation and Display of Geometric fractals in 3D", Computer Graphics 16,3 (1982) pp. 61-67.

9. *Norton, V. A.,* "Julia Sets in the quaternions, Computers and Graphics", 13, 2(1989), pp. 267-278.

10. *Peitgen, H.O., Jurgens, H. and Saupe,* D. "Chaos and Fractals", Springer-Verlag pp 837-838

11. *Pickover, C.* "Computers, Patterns, Chaos and Beauty", St. Martin's Press, pp 163-171.