

Visualizing the complex Mandelbrot trajectories

S. Halayka*

August 15, 2021

Abstract

The trajectories of the complex Mandelbrot set are visualized using Catmull-Rom curves and OpenGL.

1 Introduction

As discussed in many papers, books, and websites, a 2D scalar field of complex magnitudes (e.g. $|Z| = \sqrt{Z_x^2 + Z_y^2}$) results from calculating the complex Mandelbrot set when using a finite 2D grid of regularly spaced vertices as input. See Figure 1 for the visualization of a grid. Iteration is performed to obtain the trajectories of the complex Mandelbrot set. See Figure 2 for the tiny C++ iteration code. As shown in Figure 2, the iterative equation is

$$Z = Z^2 + C. \quad (1)$$

The criterion for an input location being in the set is that the location's trajectory's end vertex's aforementioned magnitude is always less than some threshold. See Figure 3 for a low-resolution version of the Mandelbrot set. For this paper, we use a threshold value of 4.0 for all figures.

Once all of the complex Mandelbrot set's trajectories have been generated, they are converted to Catmull-Rom curves, to be visualized using OpenGL. See Figure 4 for a visualization of some of the complex Mandelbrot

*sjhalayka@gmail.com

trajectories. See Figure 5 for a medium-resolution version of the Mandelbrot set. See Figure 6 for all of the complex Mandelbrot trajectories.

The primary motivation for the exploration of the trajectories of the complex Mandelbrot set was to introduce a new type of visualization: see Figure 7 for all of the Mandelbrot trajectories, drawn using Catmull-Rom curves using rainbow colouring.

2 Why Catmull-Rom curves?

Compared to Bézier curves, Catmull-Rom curves seem to encode a higher degree of fidelity when it comes to the line passing through all of the control points.

Catmull-Rom curves offer C_1 continuity – continuity in both position and tangent vectors. As such, the closed (periodic) loops of the complex Mandelbrot set are easy to visualize. On the other hand, Bézier curves do not offer such built-in C_1 continuity.

See Figure 8 for all of the Mandelbrot trajectories, drawn using Catmull-Rom curves and pseudorandomly-assigned colours.

Catmull-Rom curves are as computationally intensive as Bézier curves, but not much more.

Catmull-Rom curves are attractive, in several different ways.

The quaternion Mandelbrot set is also briefly considered in Figure 9.

References

- [1] Bourke. (2018) “3D volumetric fractal trajectories”
- [2] Halayka. (2018) “Visualizing the escape paths of quaternion fractals”
- [3] Chen. (2017) “C++ Bézier / spline / Catmull-Rom curve library”
<https://github.com/chen0040/cpp-spline>
- [4] Ventrella. (2021) “Mandelbrot Orbits” JavaScript Web Applet
http://www.ventrella.com/mandelbrot_orbits/

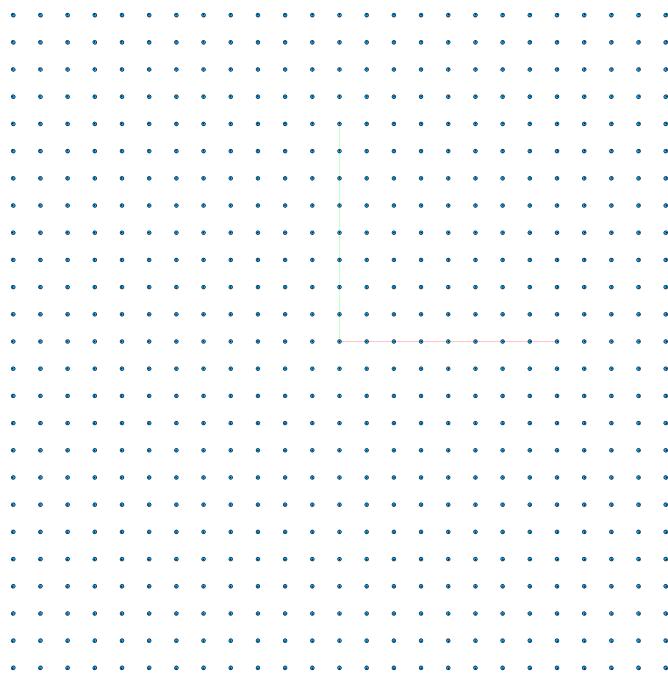


Figure 1: Grid of sample vertices. Grid minimum = -1.5, grid maximum = 1.5. Resolution = 25.

```

float iterate_mandelbrot_2d(
    vector< complex<float> & trajectory_points,
    complex<float> Z,
    complex<float> C,
    const short unsigned int max_iterations,
    const float threshold,
    const float exponent)
{
    C = Z;
    Z = complex<float>(0, 0);

    trajectory_points.clear();
    trajectory_points.push_back(Z);

    for (short unsigned int i = 0; i < max_iterations; i++)
    {
        Z = pow(Z, exponent);
        Z += C;

        trajectory_points.push_back(Z);

        if (abs(Z) >= threshold)
            break;
    }

    return abs(Z);
}

```

Figure 2: Iteration C++ code. The exponent is 2 in this paper.

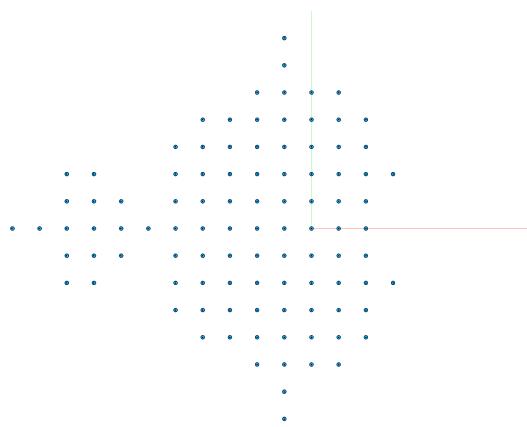


Figure 3: Low-resolution version of the complex Mandelbrot set. Only the vertices within the set are drawn. Maximum iterations = 500. Grid minimum = -1.5, grid maximum = 1.5. Threshold = 4.0. Resolution = 25.

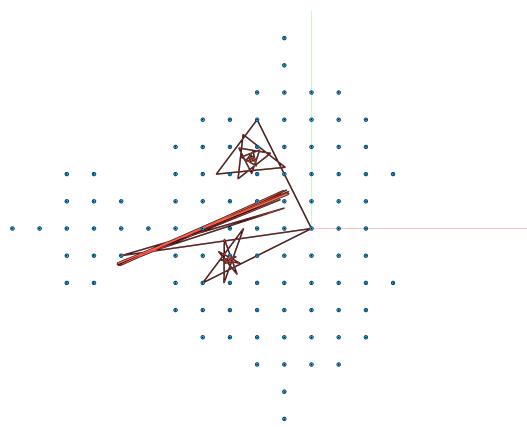


Figure 4: A few examples of the complex Mandelbrot trajectories. Maximum iterations = 500. Grid minimum = -1.5, grid maximum = 1.5. Threshold = 4.0. Resolution = 25.

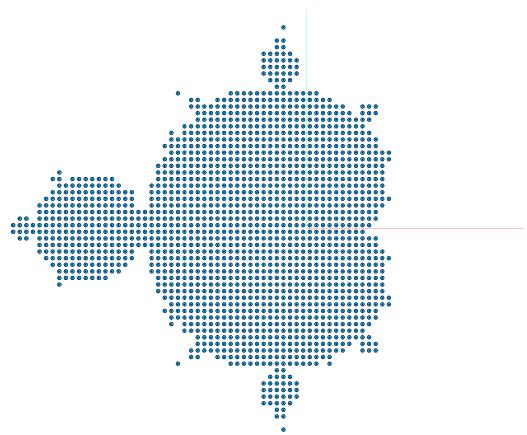


Figure 5: Medium-resolution version of the complex Mandelbrot set. Maximum iterations = 500. Grid minimum = -1.5, grid maximum = 1.5. Threshold = 4.0. Resolution = 100.

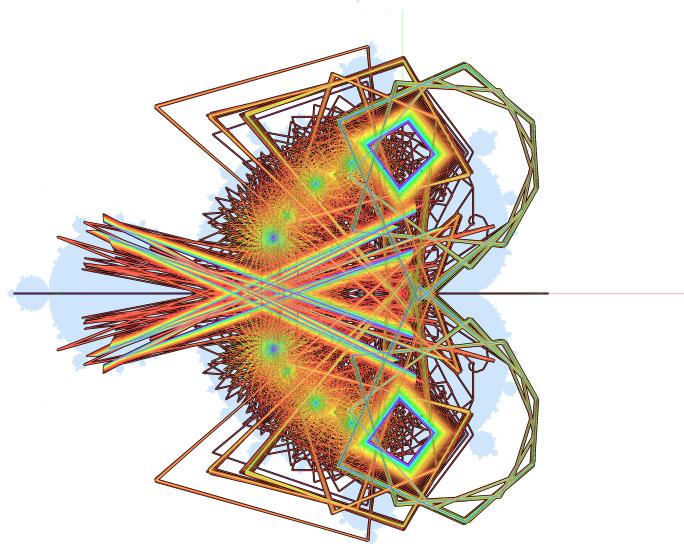


Figure 6: Complex Mandelbrot set. Maximum iterations = 500. Grid minimum = -1.5, grid maximum = 1.5. Threshold = 4.0. Resolution = 25. Actual trajectories are drawn. The majority of the trajectories end up being periodic orbits.

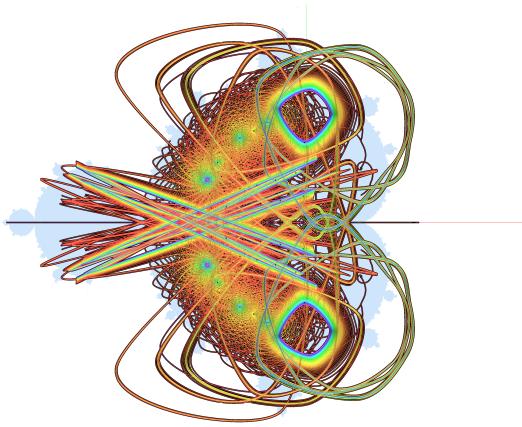


Figure 7: Complex Mandelbrot set. Maximum iterations = 500. Grid minimum = -1.5, grid maximum = 1.5. Threshold = 4.0. Resolution = 25. Catmull-Rom trajectories are drawn. The majority of the trajectories end up being periodic orbits.

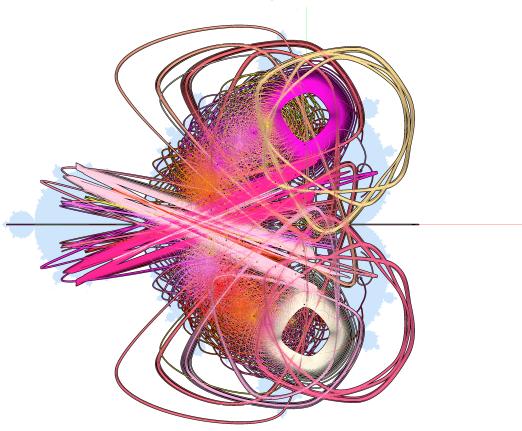


Figure 8: Complex Mandelbrot set. Maximum iterations = 500. Grid minimum = -1.5, grid maximum = 1.5. Threshold = 4.0. Resolution = 25. Catmull-Rom trajectories are drawn. Pseudorandomly-assigned colours are used, to help differentiate between the individual trajectories. The majority of the trajectories end up being periodic orbits.

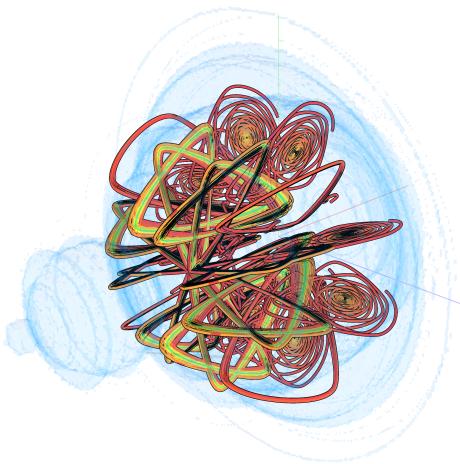


Figure 9: Quaternion Mandelbrot set. Maximum iterations = 500. Grid minimum = -1.5, grid maximum = 1.5. Threshold = 4.0. Resolution = 10. Catmull-Rom trajectories are drawn. As is with the complex Mandelbrot set, the majority of the quaternion Mandelbrot trajectories end up being periodic orbits.