

# Using Catmull-Rom curves to visualize the complex Mandelbrot trajectories

S. Halayka\*

August 8, 2021

## Abstract

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

## 1 Introduction

As discussed in many papers, 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 lattice of regularly spaced vertices as input.

Iteration is performed to obtain the trajectories of a complex Mandelbrot set. The criterion for an input location being in the set is that the location's trajectory's end vertex magnitude is always less than some threshold. For this paper, we use a maximum iteration count of 500, and a threshold value of 4.0. Once all of the Mandelbrot set's trajectories have been generated, they are converted to Catmull-Rom curves, to be visualized using OpenGL.

The primary motivation for the exploration of the trajectories of the complex Mandelbrot set was to introduce a new type of visualization.

## 2 Why Catmull-Rom 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. This is unlike Bézier

---

\*sjhalayka@gmail.com

curves, where the line is only guaranteed to go through the beginning and end control points.

Catmull-Rom curves offer  $C_1$  continuity – continuity in both position and tangent vectors. As such, the closed (periodic) loops of the Mandelbrot set are easy to visualize.

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

Catmull-Rom curves are pretty.

## References

- [1] Bourke. (2018) “3D volumetric fractal trajectories”
- [2] Halayka. (2018) “Visualizing the escape paths of quaternion fractals”

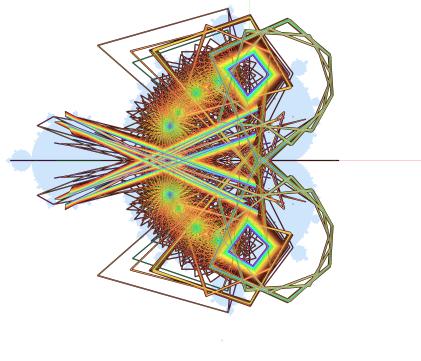


Figure 1: Iterations = 500. Actual trajectories are drawn.

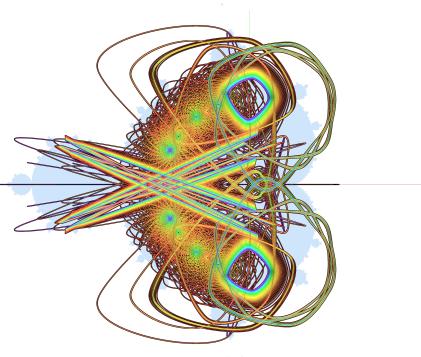


Figure 2: Iterations = 500. Catmull-Rom trajectories are drawn.