

1 Quaternions

Quaternions are four-dimensional numbers. As the complex numbers have a real component and an imaginary component, i , quaternions have a real component and three imaginary components, i, j, k . In this program, quaternions are represented with a 4-vector. Depending on the version, the data type may be float, double, long double, or some other data type which represents real values.

An example of how a quaternion is represented:

$$1 + 4i - 2j + 1.125k \cong [1, 4, -2, 1.125]$$

2 Four-dimensional space

Since the quaternions are four dimensional, navigating this space is more complicated than simply using arrow keys to move in four cardinal directions. The method I have chosen to navigate one point through this space is to assign four keys to an increase in the coordinate for four dimensions, and to assign the four keys below those to a decrease in the corresponding coordinate.

In versions 3.2 and earlier, the A-key increases the real component for the mandelbrot seed. The S-key increases the i -components of the mandelbrot seed. D and F increase the j and k components respectively. To decrease these vector components, use the Z, X, C, and V keys.

These keys increment or decrement by a variable called moveiter. If we want to move faster or slower, we can increase or decrease moveiter by using the G and B keys.

3 Changing our view in Four-dimensional space

As of version 3.2, the view is limited to a square embedded in this four-dimensional space. This means that the problem of choosing a view in our space is identical to the problem of embedding squares in this space. This is the method I have chosen to do this.

1. Choose a point, c to serve as the bottom left corner of our view square. This is a four-dimensional point, and it can be moved around with the 1,2,3,4,Q,W,E,R keys.
2. Choose a vector v to serve as the bottom edge of our view square. This is also four-dimensional, and it can be moved around with the 5,6,7,8,T,Y,U,I keys.
3. Choose a vector, o , of the same length as v , orthogonal to v , to serve as the left edge of our view. This vector is restricted to a 2-sphere, so to choose it, we will use latitude and longitude, which can be moved with the 9,0,O,P keys.

The third item may require explanation. Let V be a four-dimensional inner product space. Let $v \in V$ be the vector that we have chosen to be the bottom edge of our view. Then $W = \text{span}(v)^\perp$, the orthogonal complement of $\text{span}(v)$, is the space from which we must choose the left edge of our view. This is a

three-dimensional space, since $\dim(V) = \dim(\text{span}(v)) + \dim(\text{span}(v)^\perp)$. From W , we can only choose vectors o such that $|o| = |v|$, because we want this view to be a square. This restricts the possible values of o to be that of a surface of a sphere in W . QED.

With this fact in mind, the navigation system could apply any method at all of navigating the surface of a sphere. I have chosen the latitude and longitude method because I believe it to be intuitive, but in future versions I may implement other options.

4 Conclusion

To navigate a four-dimensional space as a point, we need four dimensions. Use the ASDFZXCV keys to navigate the mandelbrot set to choose a seed. Use the GB keys to increase or decrease how fast this point moves.

To navigate a four-dimensional space as a square, we need ten dimensions:

1. four dimensions for the bottom-left corner
2. four dimensions for the bottom-right corner
3. two dimensions for the left edge, which also fixes the other two edges.

Use the “=” and “]” keys to perform a simple zoom centered on the current view.

Use the 1234567890QWERTYUIOP keys to navigate around the julia/fatou set to choose a viewing pane. Use the “-” and “[” keys to increase or decrease how fast these points move.