Computational Methods: Project 0

Ace Chun

March 10, 2023

1 Implementations

For the implementation with fixed iterations,

```
function bisection_fixediter(func, l, r, param)
    prev_val = Float64((l + r) / 2)
    error = 1
    for i in 1:param
        temp_x = Float64((l + r) / 2)
        temp_y = func(temp_x)

        if temp_y * func(r) > 0
            r = temp_x
        elseif temp_y * func(l) > 0
            l = temp_x
        end
        cur_val = Float64((l + r) / 2)
        error = abs((cur_val - prev_val) / cur_val)
        prev_val = cur_val
    end
    return Float64((l + r) / 2), func((l + r) / 2), error
end
```

where the for loop was used to facilitate the fixed number of iterations. For the implementation to calculate an error to a certain number of decimal points,

```
function bisection_decimalplace(func, l, r, param)
    threshold = max(eps(Float64), 0.5 * 10 ^ (Float64(- param)))
    count = 0

    error = 1 # arbitrary initial error
    prev_val = Float64((l + r) / 2)

    while error > threshold
        temp_y = func((l + r) / 2)
        if temp_y * func(r) > 0
            r = (l + r) / 2
        elseif temp_y * func(l) > 0
            l = (l + r) / 2
        end
        count += 1
        cur_val = Float64((l + r) / 2)
        error = abs((cur_val - prev_val) / cur_val)
        prev_val = cur_val
    end
    return Float64((l + r) / 2), Float64(func((l + r) / 2)), count, error
end
```

We set the initial error to some arbitrary value that is nowhere close to the threshold in order to use the variable as a placeholder to begin the iterations. The threshold is defined as the maximum of the error calculated from the absolute relative approximate error theorem and machine epsilon.

2 $\sqrt[3]{3}$

To find the cube root of 3, we would need to find the roots to the equation

$$x^3 = 3 \to x^3 - 3 = 0$$

Performing the bisection method on this function with exactly three iterations, we arrive at an approximate value of

$$\sqrt[3]{3} \approx 1.4375$$

with a relative approximate error of around 0.0434.

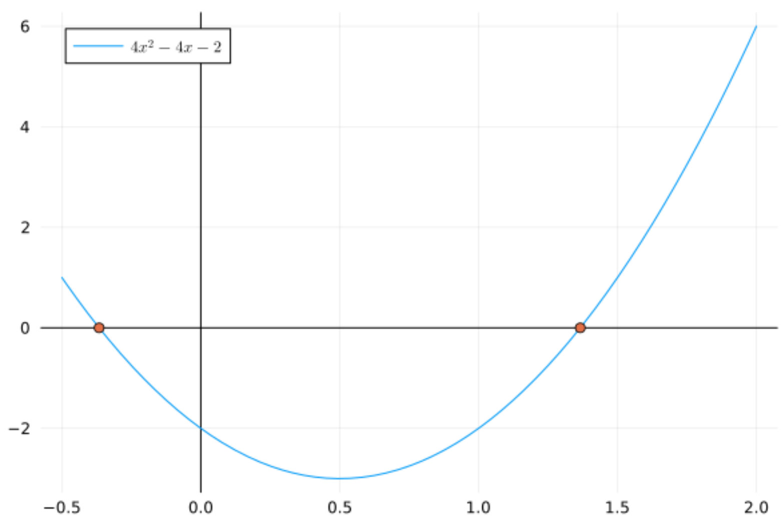
3 Polynomial

In order to find the value of

$$x = \frac{1+\sqrt{3}}{2}$$

the polynomial needed to solve this equation would be of the form

$$\begin{aligned} x &= \frac{1+\sqrt{3}}{2} \\ 2x &= 1+\sqrt{3} \\ 2x-1 &= \sqrt{3} \\ 4x^2-4x+1 &= 3 \\ \boxed{4x^2-4x-2=0} \end{aligned}$$



4 1 decimal place

Since we want to find the positive root, an interval around

$$[1, 2]$$

should be sufficient, based on the graph. Running the decimal error method from section (1) to one decimal place yields an approximation of

$$\frac{1+\sqrt{3}}{2} \approx 1.3$$

It took my function 3 iterations to achieve this approximation, and the final relative approximate error was around 0.0476.

5 3 decimal places

We run the same method, with an expectation of significant figures correct to 3 decimal places. The algorithm's final approximation for the value was

$$\frac{1+\sqrt{3}}{2} \approx 1.365$$

It took the function 10 iterations to achieve this approximation, with a final relative approximate error around 0.000357.