Mini project 3 - Fixed-Point Iteration for Intersection of Curves

Jyoutir 40360629

October 24, 2024

OBJECTIVE: Fixed-point iteration is a numerical method that finds solutions to equations of the form $x = g(x)$ by repeatedly applying the function $g(x)$ [1]. Starting from an initial guess x_0, the iteration $x_{n+1} = g(x_n)$ continues until the difference between successive approximations is smaller than a given tolerance.

This report's objective is to find the intersection point of the functions $y = xe^{x^2}$ and $y = \sqrt{1-x^2}$ in the first quadrant, defined as the region where $x \geq 0$ and $y \geq 0$, using the fixed-point iteration method. By setting $xe^{x^2} = \sqrt{1-x^2}$ and rearranging, we obtain $x = g(x)$, which is suitable for fixed-point iteration.

SUMMARY OF PROCEDURE: To find the intersection of $y = xe^{x^2}$ and $y = \sqrt{1-x^2}$ in the first quadrant, I first plotted the curves over the interval $[0, 1]$. Visual inspection of the plot suggested an initial guess, which was then refined using the fixed-point iteration method.

By equating the two functions and rearranging to express x in terms of itself, I reformulated the problem for fixed-point iteration as:

$$x = g(x) = \sqrt{\frac{1}{e^{2x^2} + 1}}, \quad x_{n+1} = g(x_n) \tag{1}$$

where x_{n+1} is the next approximation obtained from $g(x_n)$, as defined in Equation 1. The iteration continued until the absolute difference between successive approximations was below 10^{-15}, as shown in Equation 2:

$$|x_{n+1} - x_n| < 10^{-15}. \tag{2}$$

After the final approximation was obtained, I calculated the absolute errors by comparing each approximation to the final value. These errors were then plotted on a log-log scale to analyse the convergence behaviour.

Results and Discussion: Figure 1 shows the two functions, $y = xe^{x^2}$ (shown by the blue curve) and $y = \sqrt{1 - x^2}$ (shown by the orange curve), plotted over the interval $[0, 1]$. The intersection of these curves visually suggests the location of the fixed point, with an initial guess of $x_0 = 0.6$ being chosen based on this plot. This initial guess was further refined using the fixed-point iteration process.

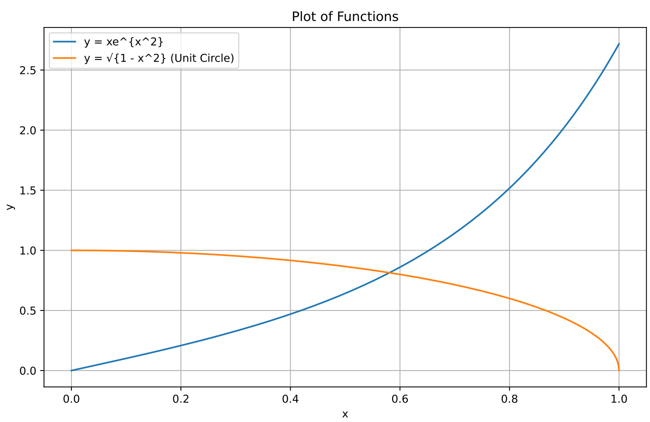

Figure 1: Intersection of $y = xe^{x^2}$ and $y = \sqrt{1 - x^2}$ in the first quadrant.

Using the fixed-point iteration method, the intersection point was found to be $x^* = 0.5808750357617376$ after 39 iterations, meeting the tolerance of 10^{-15}. Substituting x^* back into the original functions yields $y^* = x^* e^{(x^*)^2} = 0.813992747405529$, confirming the accuracy of the solution.

To further analyse the convergence behaviour, Figure 2 shows a log-log plot of the error at iteration n versus the error at iteration $n-1$. The plot shows a straight line with a slope close to 1, confirming the linear convergence of the method, which is typical of fixed-point iteration when $|g'(x^*)|$ is less than 1. In this case, $|g'(x^*)| \approx 0.4471$ (as calculated and available on GitHub [2]), indicating a linear rate of convergence. While $|g'(x^*)| < 1$ ensures convergence, the linear nature of the method means that a significant number of iterations are required to meet the desired tolerance.

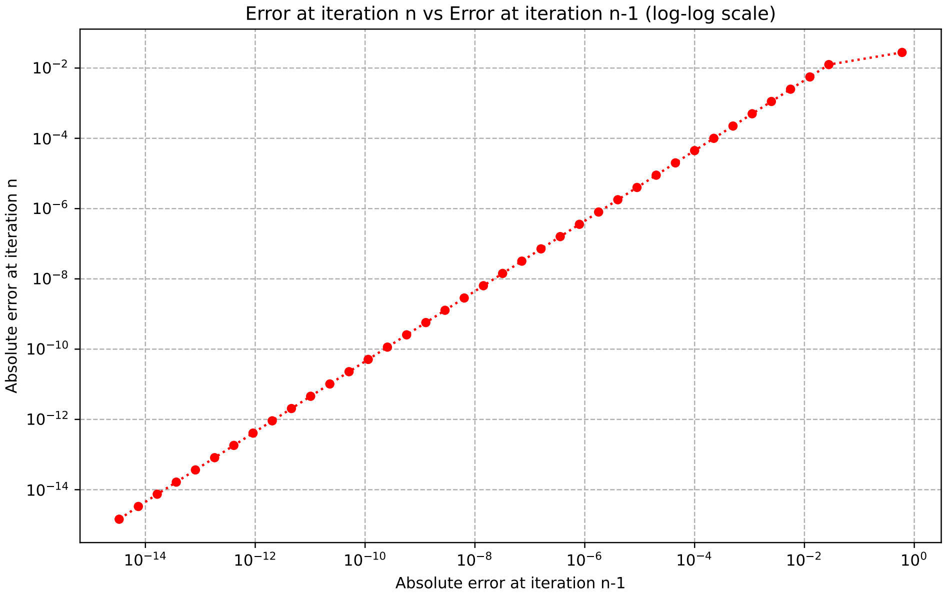

Figure 2: Log-log plot of error at iteration n versus error at iteration $n-1$ showing linear convergence.

While the fixed-point iteration method successfully found the intersection point, the linear convergence implies that further improvements could be made. One potential enhancement is to use the Newton-Raphson method, which has quadratic convergence. This method applies the iteration [3]:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, \tag{3}$$

where $f(x) = xe^{x^2} - \sqrt{1-x^2}$. The Newton-Raphson method typically converges much faster than fixed-point iteration when the initial guess is sufficiently close to the root, as the error decreases quadratically.

Conclusively, although the fixed-point iteration converged to the correct intersection point, it required 39 iterations due to the linear rate of convergence. Employing the Newton-Raphson method or acceleration techniques such as Aitken's Δ^2 method could reduce the computational cost and achieve faster convergence.

References

[1] Wikipedia. *Fixed-point iteration*. https://en.wikipedia.org/wiki/Fixed-point_iteration. Accessed: 16th October 2024.

[2] J. Raj. *Numerical Analysis Projects*. GitHub repository, https://github.com/jyoutir/numerical-analysis-projects/tree/main/miniproject_3. Accessed: 17th October 2024.

[3] Wikipedia. *Newton's method*. https://en.wikipedia.org/wiki/Newton's_method. Accessed: 17th October 2024.