Newton Raphson method

SCT211-0583/2021 Charles Daud Balila

1 Numerical root finding using Newton's method

In this document, we present a Python code for finding the root of a function using Newton's method. The code calculates the root of the function $f(x) = x^3 - 2x - 5$ with an initial guess of $x_1 = 1$, a tolerance of 10^{-6}, and a maximum of 100 iterations.

2 Python code

The following is the Python code for finding the root of the function:

```
import math
import matplotlib.pyplot as plt
import timeit

def f(x):
    return x**3 - 2*3 - 5

def df(x):
    return 3*x**2 - 2

x1 = 1
tolerance = 1e-6
max_iterations = 100

start_time = timeit.default_timer()

for i in range(max_iterations):
    fx = f(x1)
    dfx = df(x1)
    x2 = x1 - fx / dfx
    if abs(x2 - x1) < tolerance:
        print(f"Root found at x = {x2:.6f}")
        break
    else:
        x1 = x2

elapsed_time = timeit.default_timer() - start_time

print(f"Time taken to execute the code: {elapsed_time:.6f} seconds")
```

3 Explanation of the code

The code first defines the function $f(x) = x^3 - 2x - 5$ and its derivative $f'(x) = 3x^2 - 2$. It then initializes the starting point for Newton's method to $x_1 = 1$,