

Assignment4 - Submission by Team 2

June 2021

Contents

1	ME20B050 - Cecil Jacob Thomas	2
1.1	Equation - Halley's Method	2
1.2	Terms Involved	2
1.3	Description	2
1.4	Derivation	2
1.5	Applications	2
2	MM20B017 - Divya Jyothi D	3

1 ME20B050 - Cecil Jacob Thomas

1.1 Equation - Halley's Method

$$x_{n+1} = x_n - \frac{2f(x_n)f'(x_n)}{2[f'(x_n)]^2 - f(x_n)f''(x_n)} \quad (1)$$

1.2 Terms Involved

- \mathbf{x}_n : is the test value we input to the function to get the next closest approximation
- \mathbf{x}_{n+1} : is the next best approximation to the root achieved by taking x_{n+1} as the starting point
- $\mathbf{f}(\mathbf{x})$: is the function whose roots we are trying to find by finding successively better approximations to the root

1.3 Description

Halley's method also called *method of tangent hyperbolas* is a numerical method for calculating the roots of a function by producing successively better approximations to the roots of a real-valued function. It is generally more convergent than say the Newton-Rhapson method. Unlike the Newton-Rhapson method, Halley's method uses Tangent hyperbola's to approximate the next point. This method was invented by Edmond Halley

1.4 Derivation

Equation 1 can be derived by the use of a 2^{nd} order Taylor polynomial

$$y(x) = f(x_n) + f'(x_n)(x - x_n) + \frac{f''(x_n)(x - x_n)^2}{2} \quad (2)$$

Then by setting $y(x_{n+1}) = 0$ and rearranging we get:

$$x_{n+1} = x_n - \frac{2f(x_n)}{2f'(x_n) + f''(x_n)(x - x_n)} \quad (3)$$

Finally we can put in the Newton-Rhapson approximation

$x_{n+1} - x_n = -\frac{f(x_n)}{f'(x_n)}$ in the LHS to get:

$$x_{n+1} = x_n - \frac{2f(x_n)f'(x_n)}{2f'(x_n)^2 - f''(x_n)f(x_n)} \quad (4)$$

Which is essentially **Halley's root-finding algorithm**

1.5 Applications

Numerical algorithms such as these are very important as sometimes there might not exist a method to calculate an exact root by symbolic manipulation alone.

For example Kepler's equation needs to be solved many times for a variety of problems in Celestial Mechanics[1]. Therefore, computing the solution to Kepler's equation in an

efficient manner is of great importance as we could use it to solve Kepler's Equation numerically[1]. The below graph contrasts the performance of Halley's method to different root finding algorithms when applied to solve the Kepler's equation numerically for different eccentricities [1]

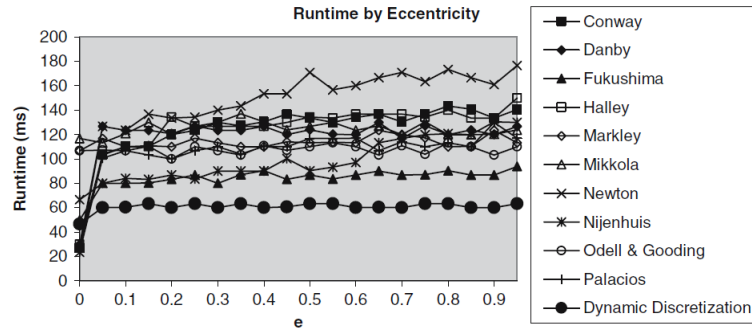


Figure 1: Runtime varying with eccentricity

Figure 1 shows us that Halley's method does certainly do better than the Newton-Rhapson method but recent techniques such as Dynamic Discretization are far more efficient

2 MM20B017 - Divya Jyothi D

References

- [1] Scott A. Feinstein and Craig A. McLaughlin. Dynamic discretization method for solving kepler's equation. *Celestial Mechanics and Dynamical Astronomy*, 96(1):49–62, Sep 2006.