MTH 308 - Mini Project (MP1)

Anurag Pandey-160140

March 30, 2019

Introduction:

The aim of the project is to design and implement (in MATLAB) a numerical scheme that solves the following problem:

Given :

- a simple closed curve C in $\mathbb{R}^2$ given by $C : [0,1] \mapsto \mathbb{R}^2$ such that

$$C(t) = (X(t), Y(t)),\ 0 \leq t \leq 1$$

  where $X, Y : \mathbb{R} \mapsto \mathbb{R}$ are infinitely differentiable periodic functions with period 1 and

- a point $(x_0, y_0) \in \mathbb{R}^2$

Find the closest point (x_c, y_c) on the curve C.

Reformulation of the problem statement

Given the outside point (x_0, y_0) we can write a distance function (f(t)) with a parameter t, which returns the square of the euclidean distance of the outside point with each point on the curve.

$$f(t) = (X(t) - x_0)^2 + (Y(t) - y_0)^2$$
$$f'(t) = X'(t) * (X(t) - x_0) + Y'(t) * (Y(t) - y_0)$$

The global minima of this $f(t)$ is our closest point (x_c, y_c). We use **secant method** to iteratively find the roots of $f'(t)$ and then check for the global minima by finding the minimum distance point among these roots.

We segment the curves in a number of parts and apply secant algorithm in all these neighbourhoods so that we do not miss on any of the roots of $f'(t)$.

Implementation details

We know that the roots of $f'(t)$ might be the local minimas instead of being the global minima. Given two initial guesses of the roots of $f'(t)$ the secant algorithm gives the root of $f'(t)$ in that neighbourhood. For Secant algorithm to converge we must give the initial guesses of the roots very close to the actual roots. Since the roots must lie for some $t \in [0, 1]$ and we are providing initial guesses with a difference of 1/1024, the root will definitely lie in between one of these 1024 segments and then the initial guess would also be quite close and hence the secant algorithm would converge.

We divide the curve in 1024 parts by diving $[0, 1]$ in uniform 1024 parts (with 1/1024 , 2/1024 being the checkpoints)and then passing the consecutive checkpoints as the initial guess of the roots of $f'(t)$ to the secant algorithm.

So we run the secant algorithm for 1024 times for a given curve,each time in one of the 1024 segments of the curve and find the roots of $f'(t)$. It might happen that in many of the segments the secant algorithm might not converge, but still it ensures that we have checked all the local properties of the curve.

For each implementation of the secant algorithm in a particular neighbourhood, we use the convergence criteria as the cosine of the angle made by the projection vector $[X(t) - x_0, Y(t) - y_0]$ with the tangent line to the curve at t, i.e. $[X'(t), Y'(t)]$ is less than the epsilon value given by the user.

We used Secant Method for finding the roots of $f'(t)$ as it has a better convergence rate over Newton's method for algebraic, trigonometric and transcendental function. Also Newton's Method requires us to calculate higher order derivatives i.e $f''(t)$ for finding the roots of $f'(t)$ which is difficult to obtain accurately.

Proof of Correctness

At the point of minimum distance say tc i.e $[X(tc), Y(tc)]$, we will have $f'(tc) = 0$. So tc is definitely among one of the roots of f'(t) .

We divide the curve in 1024 and find the roots of $f'(t)$ in each of these neighbourhoods. We check the distance of all our calculated roots with the given point and then claim a global minima. It is highly unlikely that we miss any of the roots of f'(t) for nice enough periodic functions and hence unlikely that we miss the global minima .

Thus with high probability our solution will correctly find the point on the curve with minimum distance from the outside reference point.

References

- Secant Algorithm vs Newton's Method