

16th December 2024

**Abstract**

Numerical analysis provides the foundations for solving complex systems of equations, namely differential equations which underpin our fundamental models of nature through physics. The complexity of these equations and lack of analytical solutions motivates the invention of methods which allow us to approximate solutions to these equations at high accuracy. This report looks into some simple to implement but powerful methods and how well they describe a 2nd order ODE modelling a floating object in a buoyancy model.

Numerical Methods Part 1 Report

Himmat Kaul

02376386

**Contents**

[List of Figures 2](#_Toc185274307)

[List of Tables 2](#_Toc185274308)

[List of Symbols 2](#_Toc185274309)

[Introduction 3](#_Toc185274310)

[Model Setup and Parameters 3](#_Toc185274311)

[Question 1: Analytical Solution and transformation to a first-order system of Ordinary Differential Equations 3](#_Toc185274312)

[Analytical Solution 3](#_Toc185274313)

[Transformation to first order system of ODE’s 4](#_Toc185274314)

[Question 2: Comparison of Different Numerical Schemes 4](#_Toc185274315)

[Error comparison 5](#_Toc185274316)

[Comparison of Displacements 5](#_Toc185274317)

[Velocity vs Position 6](#_Toc185274318)

[Reliability and Time complexity 7](#_Toc185274319)

[Question 3: Numerical Stability Study 7](#_Toc185274320)

[Explicit Euler 8](#_Toc185274321)

[Runge-Kutta 4th order 8](#_Toc185274322)

[Varying Initial Condition’s 8](#_Toc185274323)

[Question 4: Comparison to Spring Mass model and Introducing Damping 9](#_Toc185274324)

[Comparison to Mass Spring Model 9](#_Toc185274325)

[Damped case 9](#_Toc185274326)

[References 9](#_Toc185274327)

[Appendix 10](#_Toc185274328)

[Code 10](#_Toc185274329)

# List of Figures

[Figure 1: Analytical Solution 4](#_Toc185264035)

[Figure 2: Explicit Euler Scheme 5](#_Toc185264036)

[Figure 3: Runge-Kutta, 4th Order Scheme 5](#_Toc185264037)

[Figure 4: MATLAB™ ode45 in-built function 6](#_Toc185264038)

[Figure 5: Velocity vs Displacement for all Schemes 6](#_Toc185264039)

[Figure 6: Velocity vs Displacement for Runge-Kutta and MATLAB ode45 7](#_Toc185264040)

# List of Tables

[Table 1 Load vs. displacement 7](#_Toc185264044)

# List of Symbols

|  |  |
| --- | --- |
|  | Density of object |
|  | Density of liquid |
|  | Height of object |
|  | Cross sectional area of body |
|  | Acceleration due to gravity |
|  | Angular velocity |
|  | Time |
|  | Initial displacement |
|  | Initial Velocity |
|  | Displacement as a function of time |
|  | Acceleration as a function of time |
|  | System of functions |
|  | System of first derivatives |
|  | Matrix for System of equations |

# 

# Question 1: Analytical Solution

## Analytical Solution

The Analytical solution to the equation above can be found by considering harmonic functions as solutions, using the complex exponential:

(3)

(4)

Evaluating (1) by substituting our complex exponential,

(5)

So, our complex exponential satisfies the second order ODE and is a complete analytical solution. We are able to rewrite the complex exponential using trigonometric functions by Eulers identity:

(6)

(7)

Where , and are arbitrary constants determined by our initial conditions. Using our initial condition’s, we arrive at the constants for our model:

rad2/s2

m

Hence giving the equation and graph below for the interval ,

(8)

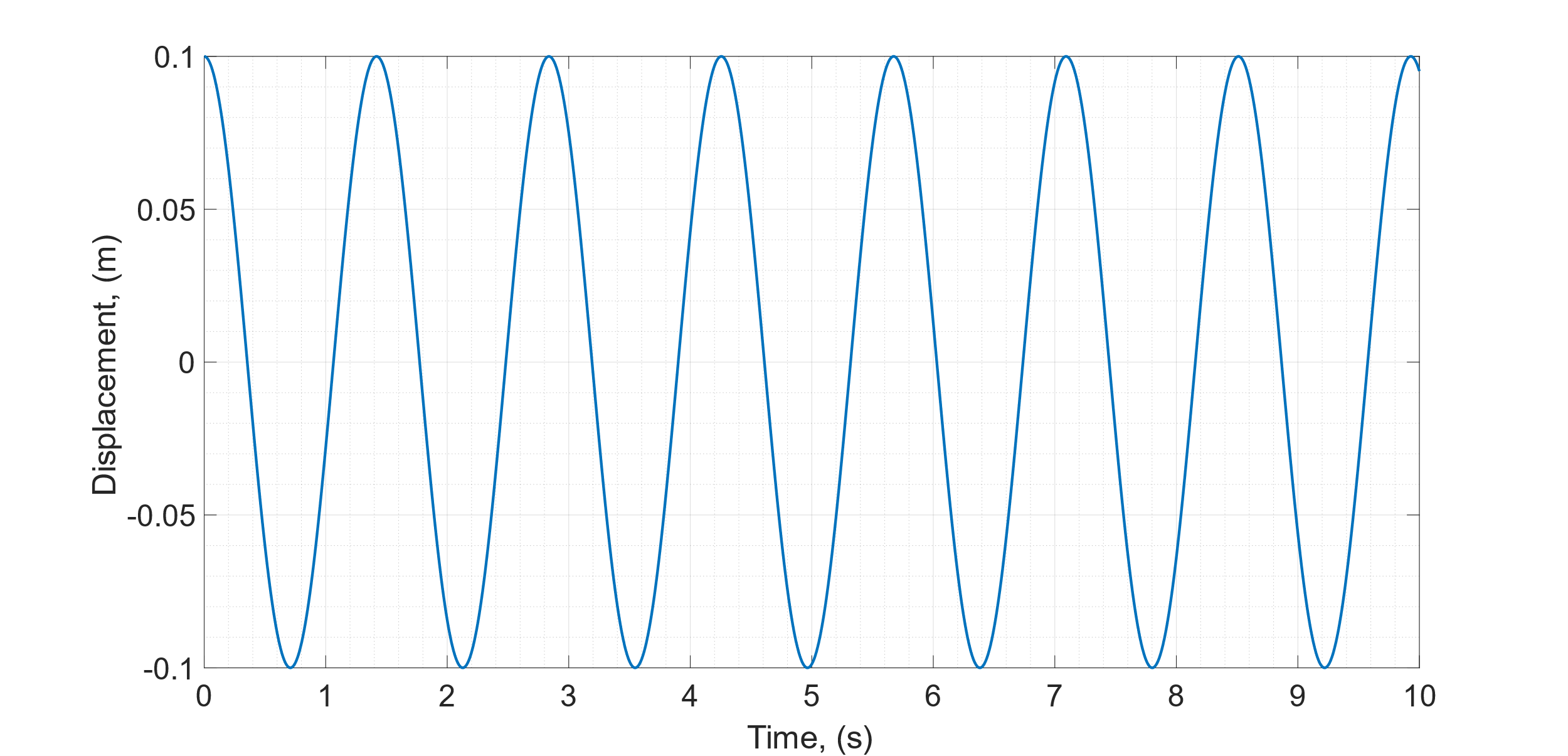


Figure 1: Analytical Solution

## Transformation to first order system of ODE’s

Using matrices we can transform equation (1) into the system of first order ODE’s using the generalised equation below using dot notation for derivatives of time.

(9)

(10)

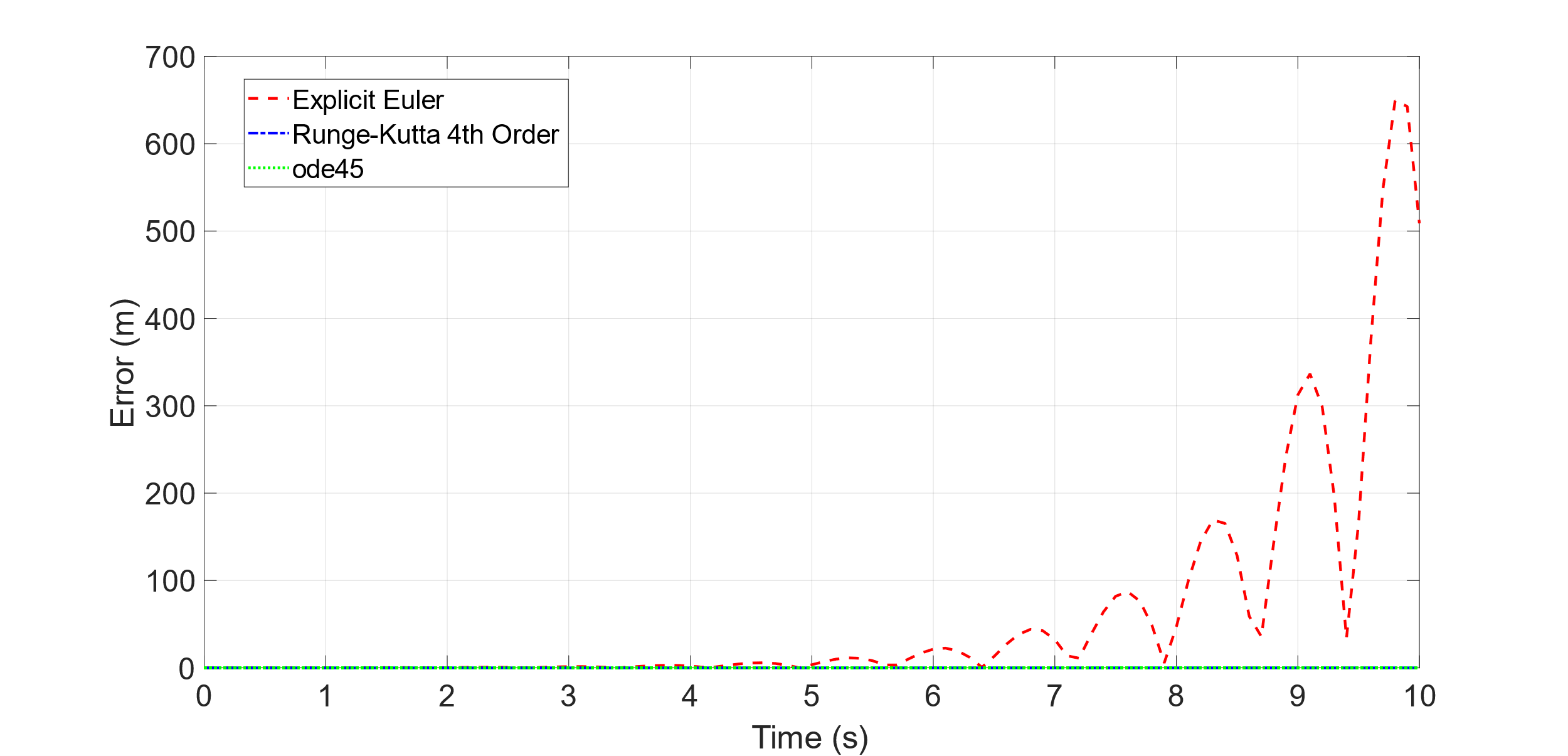
Hence the eigenvalues of the matrix can be obtained as:

,

.

# Question 2: Comparison of Different Numerical Schemes

## Error comparison



The different Numerical methods used were, Explicit Euler and Runge-Kutta 4th order, and MATLABTM ‘s inbuilt function ode45. The step size used was s, across the time interval in seconds .

## Comparison of Displacements

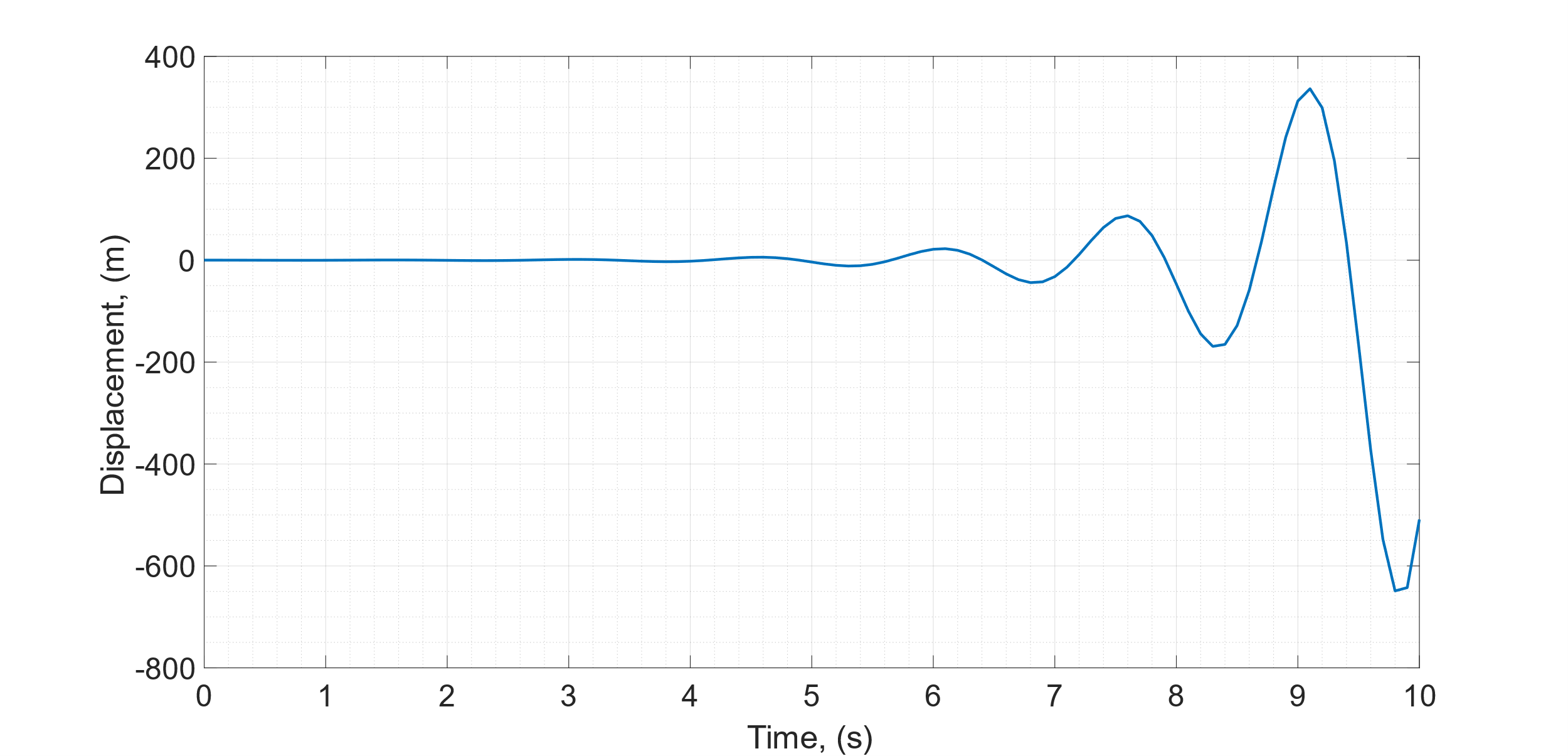


Figure 2: Explicit Euler Scheme

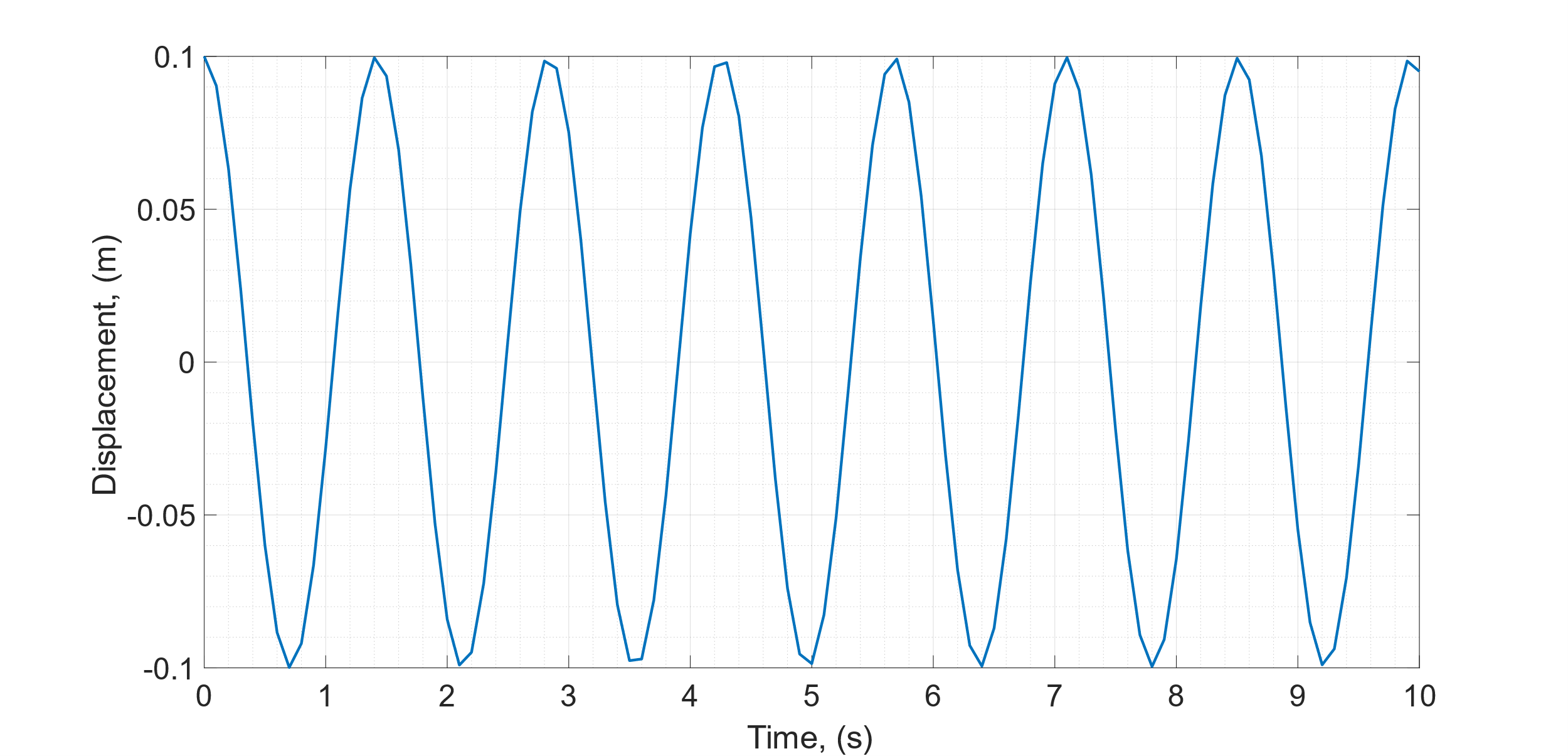


Figure 3: Runge-Kutta, 4th Order Scheme

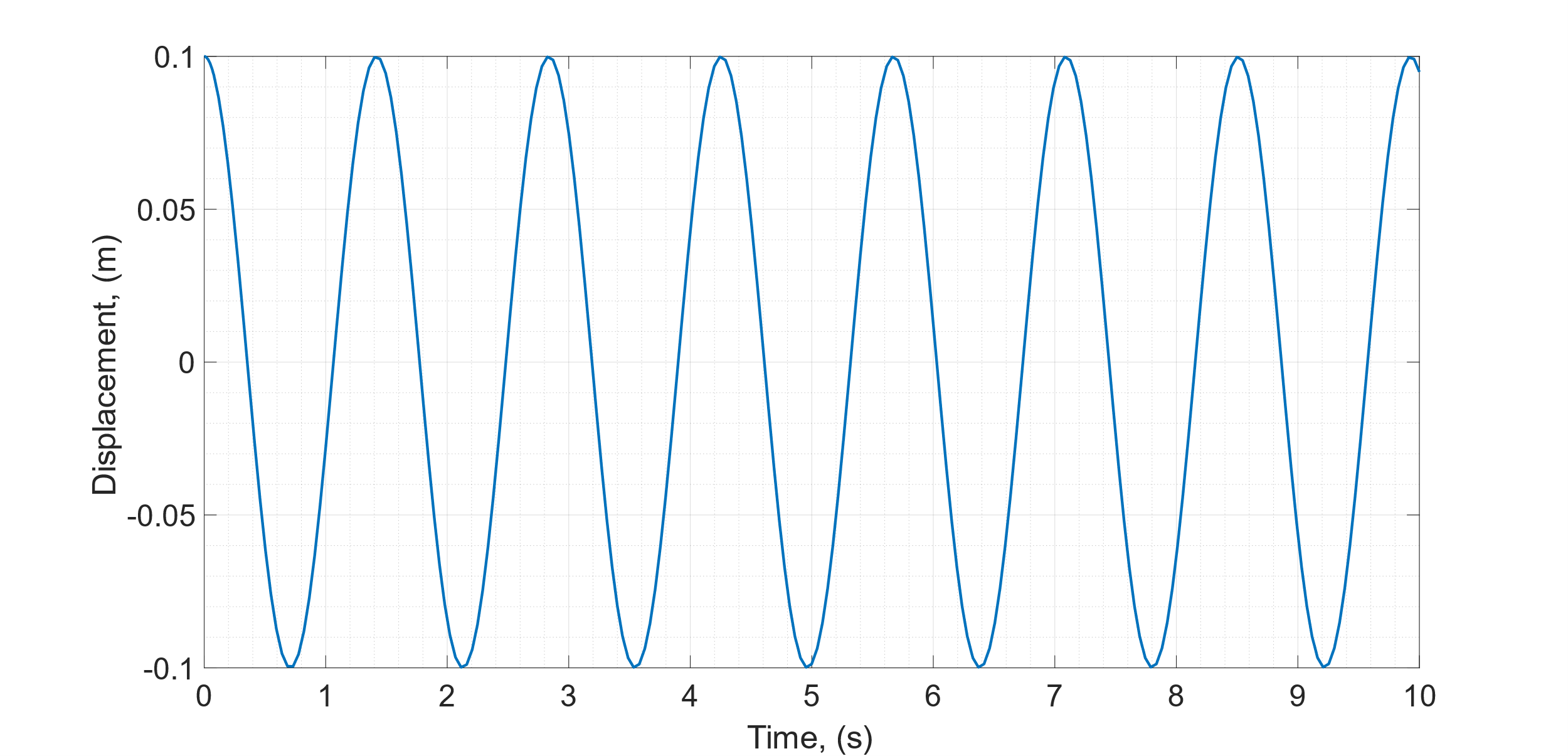


Figure 4: MATLAB™ ode45 in-built function

## Velocity vs Position

**For all schemes**

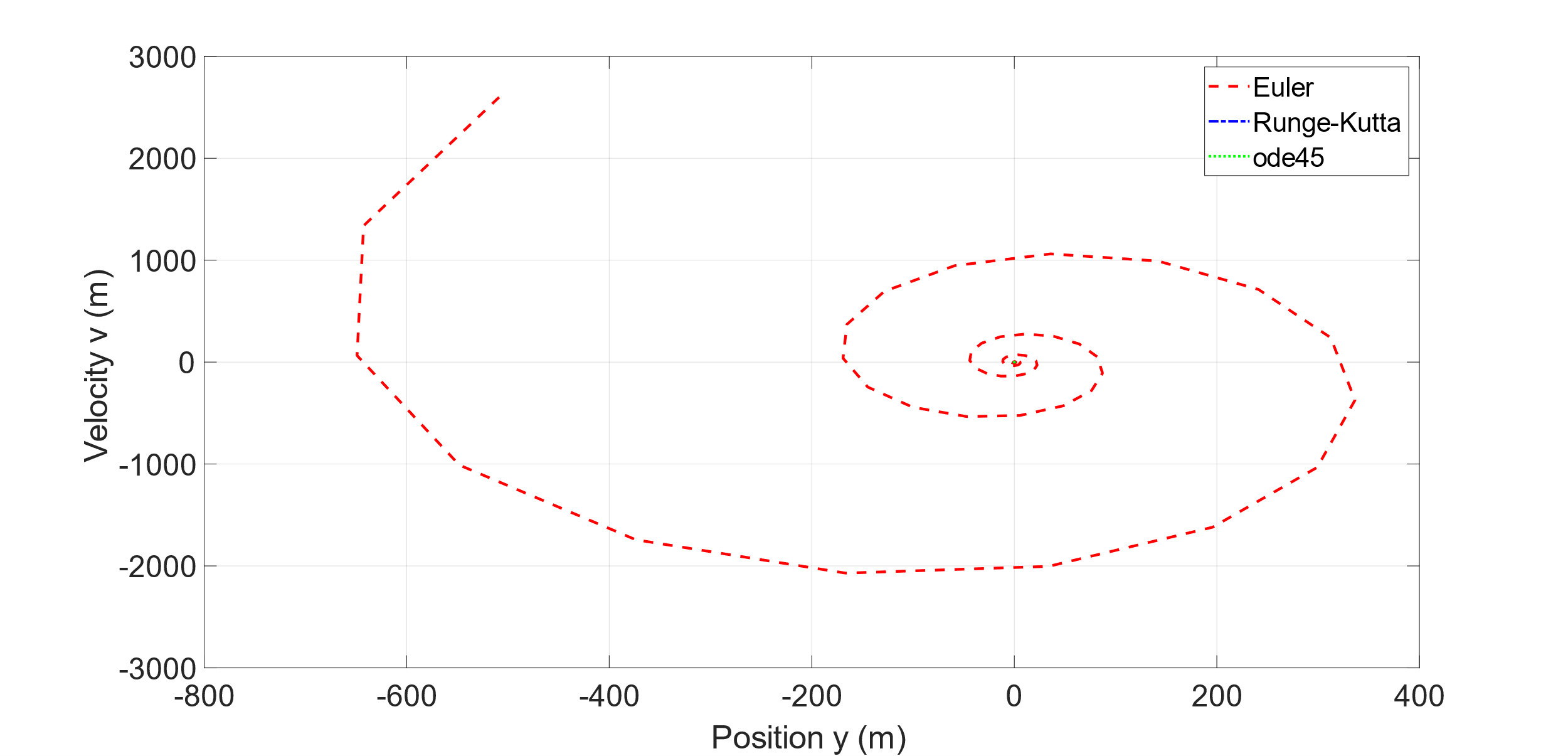


Figure 5: Velocity vs Displacement for all Schemes

**For RK4 and ode45**

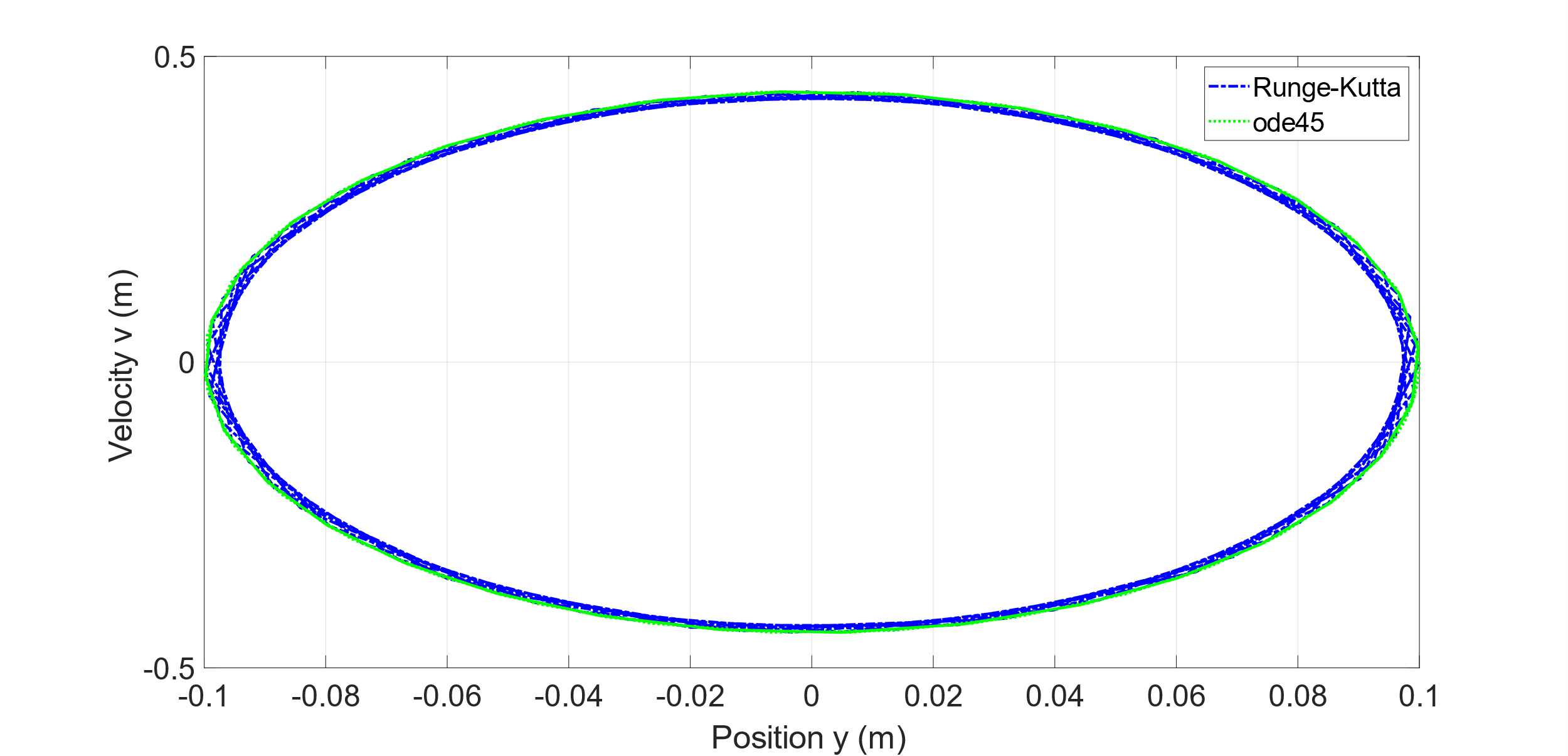
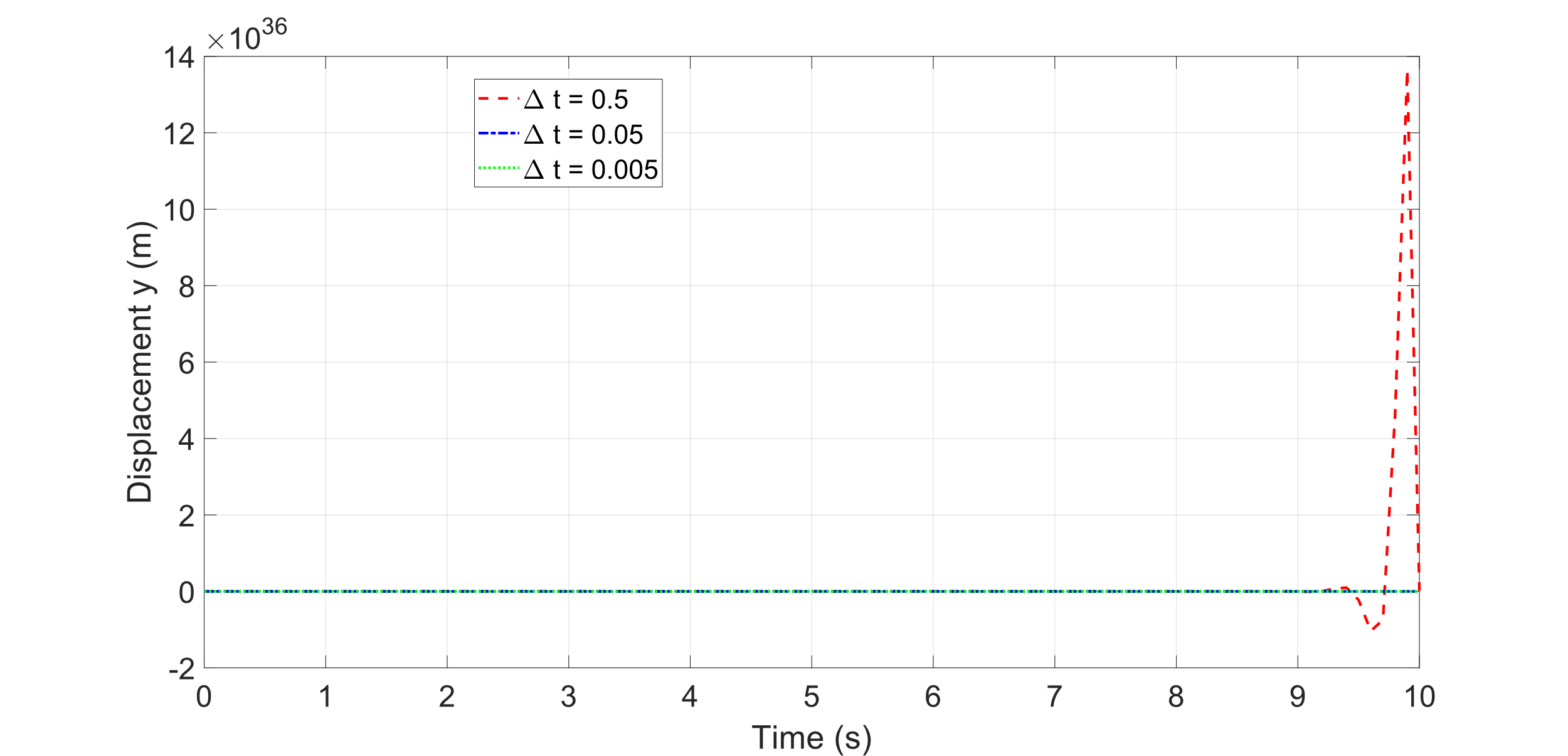


Figure 6: Velocity vs Displacement for Runge-Kutta and MATLAB ode45

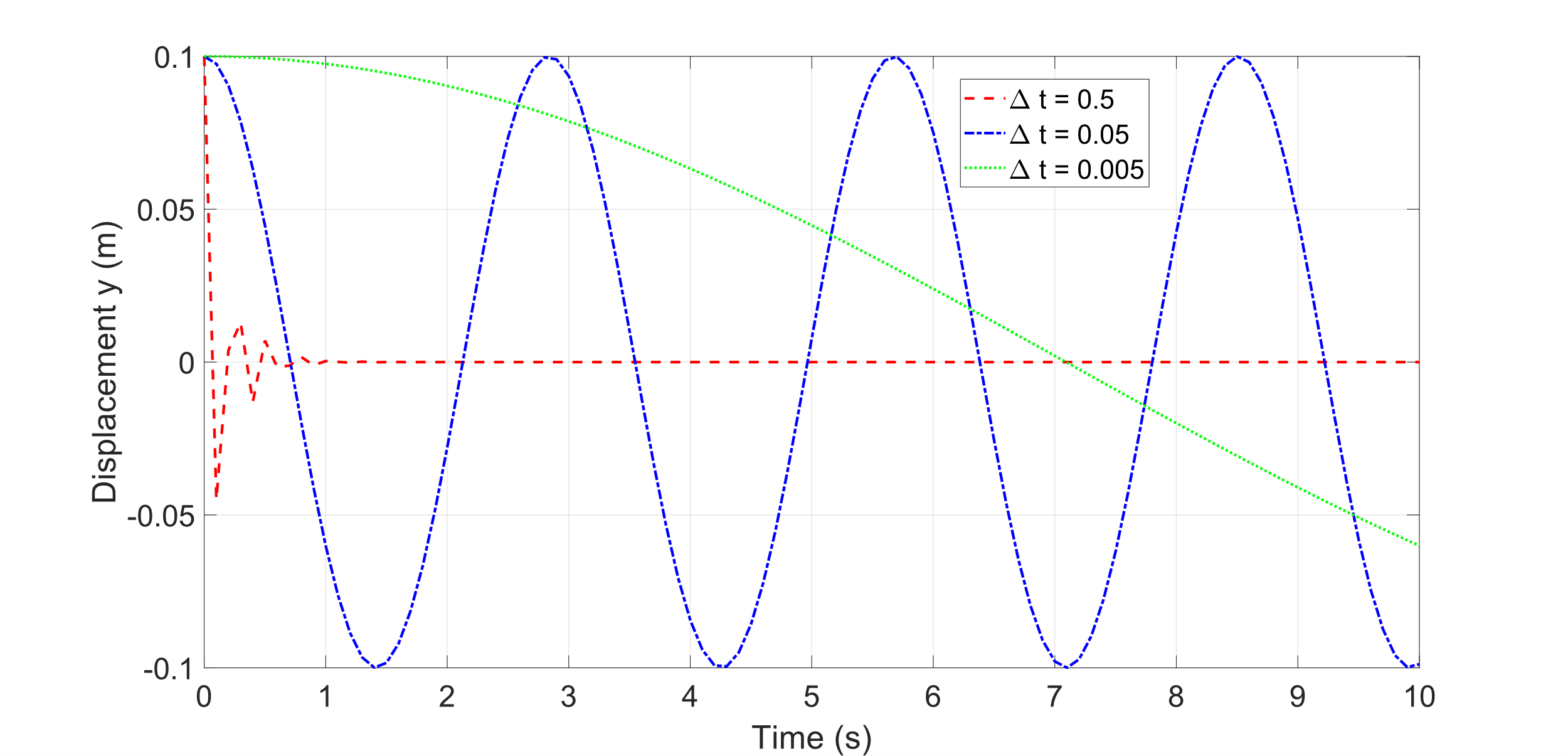
## Reliability and Time complexity

# Question 3: Numerical Stability Study

## Explicit Euler



## Runge-Kutta 4th order

****

## Varying Initial Condition’s

For further details about tables and tabulation of data, see IC Aero Report Style Guide

Table 1 Load vs. displacement

|  |  |  |
| --- | --- | --- |
| displacement (mm) | Theoretical load (KN) | Experimental load (KN) |
| 100.000 | 0.0 | 0.0 |
| 100.025 | 3.0 | 3.3 |
| 100.050 | 6.0 | 6.0 |
| 100.075 | 9.0 | 8.7 |
| 100.100 | 12.0 | 12.0 |

# Question 4: Comparison to Spring Mass model and Introducing Damping

## Comparison to Mass Spring Model

Compares to the mass spring model the buoyancy model acts in the vertical direction so have some effects of gravity which have not been included into the modelled equation. the spring mass system does not include these effects.

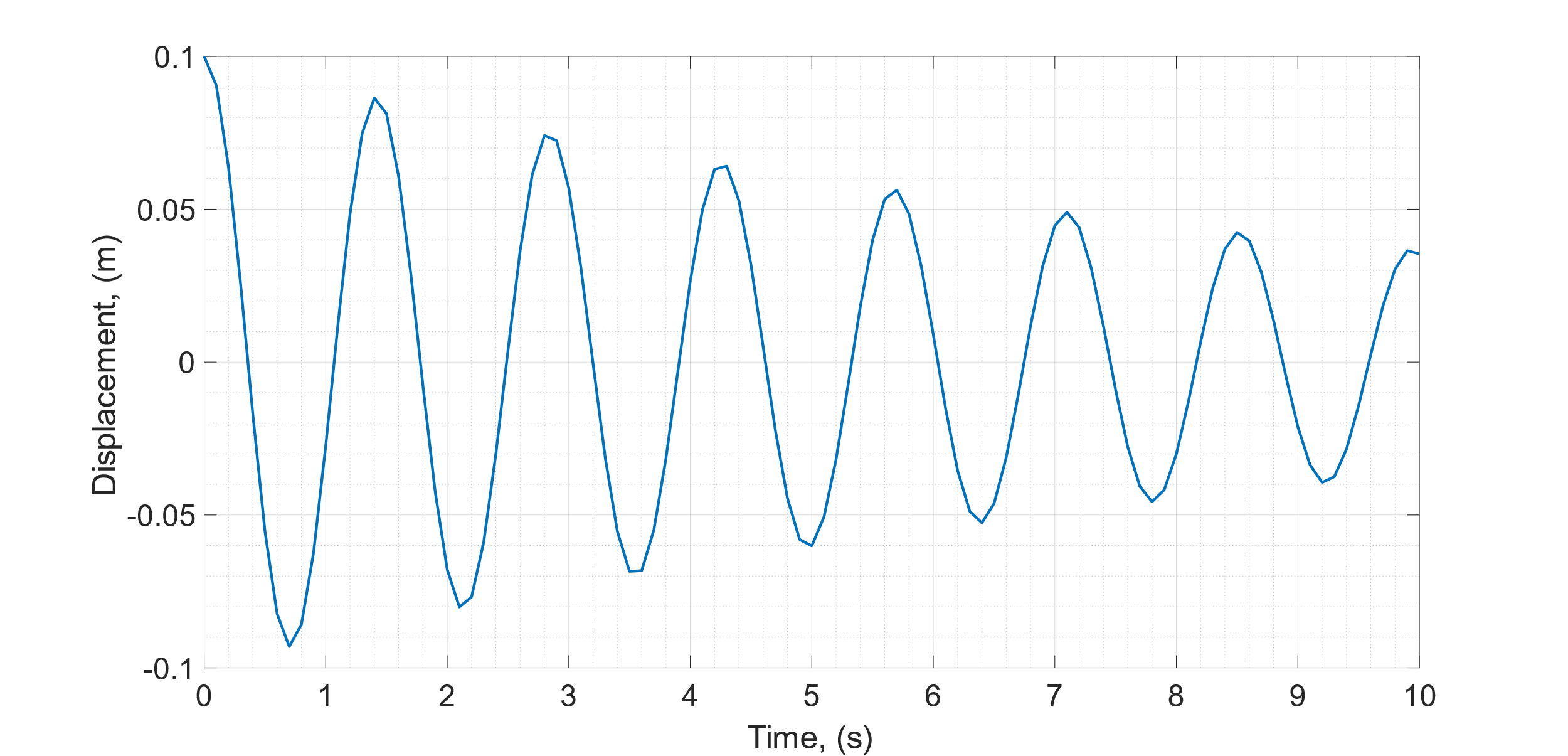
Difficulties arose when making the comparison of the Explicit Euler and the Other schemes due to instability of the time step we were using.

## Damped case

Including Damping, with c being our damping coefficient, changes our equation of motion to:

(11)

(12)



Compared to the undamped case the system is not showing a decrease in amplitude over time as opposed to the steady amplitude of the previous undamped case. The period of oscillation can also be seen to increase over time compared to the constant period for the undamped case. Introducing damping removes the first

The damped system has an increasing period and decreasing amplitude over time. However, the undamped case shows a constant time period and amplitude.

The effects of damping on numerical stability include increasing the stability of solutions due to inclusion of a dissipative mechanism which prevents runaway solutions. Additionally, it ensures solutions remain unbounded as energy is dissipated over time.

# References

[1] Please use Vancouver referencing style for all references (see IC Aero Report Style Guide).

# Appendix

## Code