PHYS 410 Homework 2

Gavin Pringle, 56401938

October 23, 2024

Introduction

Review of Theory

Numerical Approach

Implementation

Results

Conclusions

Appendix A - rk4step.m Code

```matlab
%% Problem 1 - Single Fourth Order Runge-Kutta Step

% Inputs
%       fcn:    Function handle for right hand sides of ODEs (returns
%               length-n column vector).
%       t0:     Initial value of independent variable.
%       dt:     Time step.
%       y0:     Initial values (length-n column vector).
%
% Output
%       yout:   Final values (length-n column vector)
function yout = rk4step(fcn, t0, dt, y0)
    % Compute terms in RK step
    f0 = fcn(t0, y0);
    f1 = fcn(t0 + dt/2, y0 + (dt/2)*f0);
    f2 = fcn(t0 + dt/2, y0 + (dt/2)*f1);
    f3 = fcn(t0 + dt, y0 + dt*f2);
    % Add terms to compute full RK step
    yout = y0 + (dt/6)*(f0 + 2*f1 + 2*f2 + f3);
```

Appendix B - trk4step.m Code

```
%% Problem 1 — Single Fourth Order Runge—Kutta Step

% Inputs
%       fcn:    Function handle for right hand sides of ODEs (returns
%               length—n column vector).
%       t0:     Initial value of independent variable.
%       dt:     Time step.
%       y0:     Initial values (length—n column vector).
%
% Output
%       yout:   Final values (length—n column vector)
function yout = rk4step(fcn, t0, dt, y0)
    % Compute terms in RK step
    f0 = fcn(t0, y0);
    f1 = fcn(t0 + dt/2, y0 + (dt/2)*f0);
    f2 = fcn(t0 + dt/2, y0 + (dt/2)*f1);
    f3 = fcn(t0 + dt, y0 + dt*f2);
    % Add terms to compute full RK step
    yout = y0 + (dt/6)*(f0 + 2*f1 + 2*f2 + f3);
```