Efficiency Evaluation of Forward Euler Method-Based Numerical Methods for Approximating 2nd-order Ordinary Differential Equations

JQX051[1]

Higher Level Mathematics Analysis and Approaches

I. Introduction

In writing an extended essay on airfoil efficiency optimization, I used a computational fluid dynamics solver (CFD) to approximate forces exerted on rigid bodies by fluids. Fundamentally, the program solved ordinary and partial differential equations of varying orders among other intermediary processes. Each iteration required considerable computational capacity and time, and thus, finding the most efficient numerical method for solving ordinary differential equations (ODEs) would allow processes like CFD solving to be optimized. The efficiency of a method, as defined for this exploration, is maximized when complexity—measured through the number of operations runtime—is minimized. Efficiency metrics evaluated in this exploration include convergence rate, runtime-discretization ranges, and runtime-error ratios.

II. Numerical Methods

Three ground-level approaches to approximating second order ODEs will be evaluated. Namely, the Euler method, the Runge-Kutta midpoint (RK2) method, and the multistep predictor-corrector (P-C) method. These methods provide varying approaches for computing solutions to differential equations, as will be detailed, which enables effective efficency comparisons.

In describing each method, the second-order ODE describing the simple harmonic motion of a mass-pendulum system,

$$\frac{\mathrm{d}^2x}{\mathrm{d}t^2} = -\omega^2 x, \tag{1}$$

will be used for sample calculations. For all methods, the equation must be decomposed into a system of first-order ODEs:

[1]

$$\begin{aligned} \frac{\mathrm{d}x}{\mathrm{d}t} &= v \\ \frac{\mathrm{d}^2x}{\mathrm{d}t^2} &= \frac{\mathrm{d}v}{\mathrm{d}t} = -\omega^2 x \end{aligned} \tag{2}$$

The samples calculations will consider the initial conditions $\omega = \sqrt{10}$, $x_0 = 1$, $v_0 = 0$, and $h = 0.01$ where ω is the angular frequency of the system, x_0 is the initial displacement of the mass from equilibrium, v_0 is the initial velocity of the mass, and h is the iterative step size.

A. Forward Euler Method

The forward Euler method is an explicit method that uses the slope at the point to approximate the function's value at the next point. To find x_{n+1} and v_{n+1} at time $t_{n+1} = t_n + h$, the following equations are forumlated:

$$\begin{aligned} v_{n+1} &= v_n + h \times -\omega^2 x_n \\ x_{n+1} &= x_n + h \times v_n. \end{aligned} \tag{3}$$

Given the prescribed initial conditions, the ODE can be approximated with the foreward Euler method as follows:

$$\begin{aligned} v_{n+1} &= 0 + 0.01 \times -10 = -0.1 \\ x_{n+1} &= 1 + 0.01 \times 0 = 1. \end{aligned} \tag{4}$$

To predict the motion of the pendulum system over time, these final values are used to approximate the next iteration. The RK2 method involves 2 main calculations per iteration for 2nd order ODEs, and thus, the number of operations is proportional to the number of iterations.

B. Runge-Kutta Midpoint (RK2) Method

The Runge-Kutta Midpoint (RK2) method is a numerical method for computing 2nd order ODEs based on the higher order RK4 method. RK2 adopts a similar approach to the foreward Euler method, but introduces a half step (midpoint) between iterations to improve the accuracy of the approximations, as illustrated in Fig. 1.

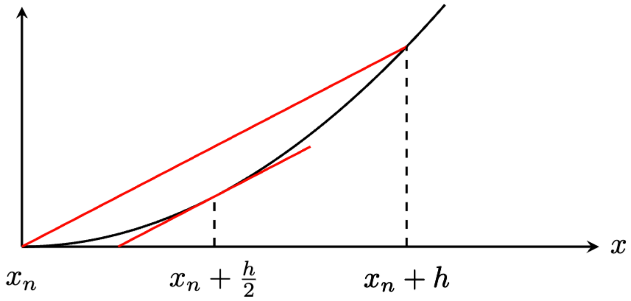

Fig. 1 RK2 Method

To find x_{n+1} and v_{n+1} at time $t_{n+1} = t_n + h$, the derivative k of the initial point is used to calculate values at midpoint m:

$$\begin{aligned} k1_v &= -\omega^2 x_n \\ k1_x &= v_n \\ m_v &= v_n + \frac{1}{2}h \times k1_v \\ m_x &= x_n + \frac{1}{2}h \times k1_x. \end{aligned} \tag{5}$$

The slopes at the midpoint are then used to calculate the final values:

$$\begin{aligned} k2_v &= -\omega^2 m_x \\ k2_x &= m_v \\ x_{n+1} &= x_n + h \times k2_x \\ v_{n+1} &= v_n + h \times k2_v. \end{aligned} \tag{6}$$

Given the prescribed initial conditions, the ODE can be approximated with RK2 as follows, starting with the half step,

$$\begin{aligned} k1_v &= -10 \times 1 = -10 \\ k1_x &= 0 \\ m_v &= 0 + \frac{1}{2} \times 0.01 \times (-10) = -0.05 \\ m_x &= 1 + \frac{1}{2} \times 0.01 \times 0 = 1, \end{aligned} \tag{7}$$

followed by the full step,

$$\begin{aligned} k2_v &= -10 \times 1 = -10 \\ k2_x &= -0.05 \\ x_{n+1} &= 1 + 0.01 \times (-0.05) = 0.9995 \\ v_{n+1} &= 0 + 0.01 \times (-10) = -0.1. \end{aligned} \tag{8}$$

The RK2 method involves 8 main calculations per iteration for 2nd order ODEs, and thus, the number of operations is proportional to the number of iterations.

C. Multistep Predictor-Corrector Method (P-C)

The predictor-corrector (P-C) method is a multistep explicit method involving a predictor step, in which the values are predicted using the foreward Euler method, then adjusted in the corrector step using the Adams-Bashforth

Method. To find x_{n+1} and v_{n+1} at time $t_{n+1} = t_n + h$, the foreward Euler method is applied to predict v and y at time t_{n+1}:

$$\begin{aligned} v_{\text{prediction}} &= v_n + h \times -\omega^2 x_n \\ x_{\text{prediction}} &= x_n + h \times v_n. \end{aligned} \tag{9}$$

This is followed by the corrector step, which is applied conditionally. For the first iteration where a previous v value is unavailable,

$$v_{\text{corrected}} = v_{\text{prediction}}. \tag{10}$$

Otherwise, for later iterations where a previous v value is available,

$$v_{\text{corrected}} = v_n + h\left(\frac{3}{2}(-\omega^2 x_n) - \frac{1}{2}(-\omega^2 x_{n-1})\right). \tag{11}$$

In both cases, $x_{\text{corrected}}$ is calculated as

$$x_{\text{corrected}} = x_n + \frac{h}{2}(v_n + v_{\text{corrected}}). \tag{12}$$

For the computation of following iterations, the following assignments can be made:

$$\begin{aligned} x_{n+1} &= x_{\text{corrected}} \\ v_{n+1} &= v_{\text{corrected}}. \end{aligned} \tag{13}$$

Given the prescribed initial conditions, the ODE can be approximated with P-C as follows, starting with the prediction,

$$\begin{aligned} v_{\text{prediction}} &= 0 + 0.01 \times (-10 \times 1) = -0.1 \\ x_{\text{prediction}} &= 1 + 0.01 \times 0 = 1, \end{aligned} \tag{14}$$

followed by the corrector,

$$\begin{aligned} v_{n+1} &= -0.1 \quad \text{(since it is the first step)} \\ x_{n+1} &= 1 + \frac{0.01}{2}(0 - 0.1) = 0.9995. \end{aligned} \tag{15}$$

To predict the motion of the pendulum system over time, these final values are used to approximate the next iteration following Equation (11). The P-C method involves 4 main calculations per iteration for 2nd order ODEs.

III. Efficiency Evaluation

The efficiency of each numerical method can be quantified by via computational metrics such as its convergence rate, its number of operations, optimal discretization ranges, and operation to error ratios. To evaluate each metric, each numerical method was applied to approximate two real-world initial value problems (IVP): The equation of motion of a simple harmonic oscillator (IVP1),

$$\frac{\mathrm{d}^2x}{\mathrm{d}t^2} = -\omega^2 x, \tag{16}$$

and the equation of motion of a damped harmonic oscillator (IVP2),

$$\frac{\mathrm{d}^2x}{\mathrm{d}t^2} + b\frac{\mathrm{d}x}{\mathrm{d}t} + kx = 0. \tag{17}$$

These 2nd order IVPs were selected for their linearity and homogeneity, allowing analytical solutions to be derived[†].

A. Convergence Rate

The convergence rate of a method is defined by its rate of reduction in error with respect its step size h. A suitable error metric for examining convergence is the root-mean-square-error (RMSE). The metric uses the same scale as the target variable, enabling a maximum absolute error threshold to be defined—a crucial consideration for computing physical quantities. RMSE quantifies the deviation between analytical and numerical solutions. Thus, to calculate it, analytical solutions are derived for comparison, with the first (IVP1) using the standard second-order homogeneous ODE form:

$$\begin{gathered}\frac{\mathrm{d}^2x}{\mathrm{d}t^2} = -\omega^2 x \\ \frac{\mathrm{d}^2x}{\mathrm{d}t^2} + \omega^2 x = 0\end{gathered} \tag{18}$$

letting $x'' = r^2$, we have

$$\begin{gathered}r^2 = -\omega^2 \\ r = \pm\omega i.\end{gathered} \tag{19}$$

Because r has complex conjugate roots in the form of $\alpha \pm \beta i$, its general solution is in the form

$$\begin{gathered}x(t) = e^{\alpha t}[C_1\cos(\beta t) + C_2\sin(\beta t)], \\ \therefore x(t) = C_1\cos(\omega t) + C_2\sin(\omega t).\end{gathered} \tag{20}$$

Considering the simple harmonic motion initial conditions $v_0 = 0$ and $x_0 = 1$, when we let $x(0) = x_0$, the cosine term is 1 and the sine term is 0, therefore, $C_1 = x_0$. Likewise, given $\frac{\mathrm{d}x}{\mathrm{d}t}(0) = v_0$,

[†] Test scripts and initial conditions are in the appendix.

$$
\begin{aligned}
\frac{\mathrm{d}x}{\mathrm{d}t} &= -\omega C_1 \sin(\omega t) + \omega C_2 \cos(\omega t), \\
\therefore v_0 &= -\omega C_1 \sin(0) + \omega C_2 \cos(0) \\
v_0 &= \omega C_2 \\
C_2 &= \frac{v_0}{\omega}.
\end{aligned}
\tag{21}
$$

Thus, we arrive at the analytical solution

$$
\boxed{x(t) = x_0 \cos(\omega t) + \frac{v_0}{\omega} \sin(\omega t).} \tag{22}
$$

IVP2 is solved similarly using the standard second-order homogeneous ODE form:

$$
\frac{\mathrm{d}^2 x}{\mathrm{d}t^2} + b\frac{\mathrm{d}x}{\mathrm{d}t} + kx = 0. \tag{23}
$$

letting $x'' = r^2$, we have

$$
\begin{aligned}
r^2 + br + k = 0, \\
\therefore r = \frac{-b \pm \sqrt{b^2 - 4k}}{2}.
\end{aligned}
\tag{24}
$$

For $b^2 - 4k > 0$, the following generalization can be made for computation when b and k are known, based on the general solution form:

$$
\begin{aligned}
x(t) &= C_1 e^{r_1 t} + C_2 e^{r_2 t} \\
r_1 &= -b + \sqrt{b^2 - 4k} \\
r_2 &= -b - \sqrt{b^2 - 4k}.
\end{aligned}
\tag{25}
$$

Applying the initial conditions $v_0 = 0$ and $x_0 = 1$,

$$
\begin{aligned}
1 &= C_1 + C_2 \\
0 &= C_1 r_1 + C_2 r_2, \\
\therefore C_2 &= x_0 - C_1, \\
\therefore C_1 &= \frac{v_0 - r_2 x_0}{r_2 - r_2}.
\end{aligned}
\tag{26}
$$

Thus,

$$
\boxed{x(t) = \frac{v_0 - r_2 x_0}{r_2 - r_2} \cdot e^{-b+\sqrt{b^2-4k}\cdot t} + \left(x_0 - \frac{v_0 - r_2 x_0}{r_2 - r_2} \right) \cdot e^{-b-\sqrt{b^2-4k}\cdot t}.} \tag{27}
$$

For $b^2 - 4k < 0$, the general solution is in the form

$$
x(t) = e^{\alpha t}[C_1 \cos(\beta t) + C_2 \sin(\beta t)]. \tag{28}
$$

Applying the initial conditions $v_0 = 0$ and $x_0 = 1$,

$$1 = e^0[C_1 \cos(0) + C_2 \sin(0)] = C_1 \tag{29}$$

and

$$\begin{aligned}
\frac{\mathrm{d}x}{\mathrm{d}t} &= \alpha e^{\alpha t}[C_1 \cos(\beta t) + C_2 \sin(\beta t)] + e^{\alpha t}[-\beta C_1 \sin(\beta t) + \beta C_2 \cos(\beta t)] \\
&= e^{\alpha t}[(\alpha C_1 - \beta C_2) \cos(\beta t) + (\alpha C_2 - \beta C_1) \sin(\beta t)] \\
\therefore 0 &= e^0[(\alpha C_1 - \beta C_2) \cos(0) + (\alpha C_2 - \beta C_1) \sin(0)] \\
0 &= \alpha C_2 - \beta C_1.
\end{aligned} \tag{30}$$

Since $x_0 = C_1$,

$$C_2 = \frac{v_0 - \alpha x_0}{\beta} \tag{31}$$

Thus, given $\alpha = -\dfrac{b}{2}$ and $\beta = \dfrac{\sqrt{4k - b^2}}{2}$ from the complex conjugate roots of r in the form $\alpha + \beta i$,

$$\boxed{x(t) = e^{-\frac{1}{2} bt} \left[x_0 \cos\left(\frac{\sqrt{4k - b^2}}{2} \cdot t \right) + \frac{2v_0 + bx_0}{\sqrt{4k - b^2}} \sin\left(\frac{\sqrt{4k - b^2}}{2} \cdot t \right) \right].} \tag{32}$$

Using these analytical solutions, the

B. Complexity Range

C. Runtime-Error

IV. Accuracy Evaluation

- for SHM and damped equations: plot against analytical solution
- plot absolute error
- global error analysis: RMSE
- analyse accuracy via convergence rate:
- run method w different stepsizes (magnitudes of 2)
- calculate rmse of each step size
- analyze error reductiona nd calc convergence rate (convergence rate formula)

$$p \approx \frac{\log\left(\frac{E(h_1)}{E(h_2)}\right)}{\log\left(\frac{h_1}{h_2}\right)} \tag{33}$$

compare convergence rate across methods higher order methods = higher efficiency

results:

- for SHM, only RK2 converges, while the other methods increase in abs. in an increasing oscillating pattern for every iteration

- for damped, all solutions converge. RK2 still converges the fastest, but P-C and Reverse Euler vary between test cases.
- visual comparison of graphs (global)

V. Efficiency Evaluation

RK2 is the best overall

Reverse Euler is lowest performing

P-C is quite accurate across the board and can be used for stiffer equations

Defining Efficiency: accuracy/time -> comparison of this figure leads to RK2 being the most efficient.

Important consideration + extension: Reverse euler is much more stable, so it performs well with larger step sizes. This was not changed as it was a control, but it greatly affects its performance as its runtime is comparable to that of the other methods at large step sizes of up to h=3, which is 100x more than the tested size. This does not work for the other two methods.

- this may be considered and explored for the final submission, if advised.

Extension: Test step sizes for full optimization.

Note: "optimization" is better done within method classes themselves. This exploration only covers and outlines the differences between each method class. Focusing on optimizing one solution/method will allow for proper modeling of step size vs. time vs. convergence rate for full optimization. "optimization" in this case is more like "evaluation", which was an oversight when this exploration was planned: method types cannot be quantified for proper optimization.

Thus, the paper title may be changed to match that: Efficiency Evaluation…

note to Examiner: All the data has been collected, and all of the code and graphs have been generated. I will work to add all of these results for the final submission.

See the full source code at https://github.com/perakasem/hlaa-ia

- contains code for each numerical method: will be added to appendix for final submission.

VI. Conclusion

References

[1] Burkardt, J., "The Midpoint and Runge Kutta Methods", Jul 13 2011.

[2] P. Kavitha, and K. Prathiba, "Adams-Bashforth Corrector-Predictor Method Using MATLAB", International Journal of Mechanical Engineering, Kalahari Journals, Apr2022.. Retrieved 8 January 2024. https://kalaharijournals.com/resources/APRIL_93.pdf

[3] Brorson, S., "Backward Euler Method".