Efficiency Optimization of Numerical Methods for Approximating Second Order Ordinary Differential Equations.

Pera Kasemsripitak[1]

NIST International School, Bangkok, 10110, Thailand

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem.

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 runtime—is minimized, while the accuracy of the results is maximized.

II.   Numerical Methods

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

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

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

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

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

The samples 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.

---

[1]Student, International Baccalaureate Diploma.

A. 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. In essence, RK2 operates similar to the Euler method, but introduces a half step (midpoint) between iterations to improve the accuracy of the approximations. 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:

$$k1_v = -\omega^2 x_n$$
$$k1_x = v_n$$
$$m_v = v_n + \frac{1}{2}h \times k1_v \qquad (3)$$
$$m_x = x_n + \frac{1}{2}h \times k1_x.$$

The midpoint values are then used to evaluate the final values:

$$k2_v = -\omega^2 m_x$$
$$k2_x = m_v$$
$$x_{n+1} = x_n + h \times k2_x \qquad (4)$$
$$v_{n+1} = v_n + h \times k2_v.$$

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

$$k1_v = -10 \times 1 = -10$$
$$k1_x = 0$$
$$m_v = 0 + \frac{1}{2} \times 0.01 \times (-10) = -0.05 \qquad (5)$$
$$m_x = 1 + \frac{1}{2} \times 0.01 \times 0 = 1,$$

followed by the full step,

$$k2_v = -10 \times 1 = -10$$
$$k2_x = -0.05$$
$$x_{n+1} = 1 + 0.01 \times (-0.05) = 0.9995 \qquad (6)$$
$$v_{n+1} = 0 + 0.01 \times (-10) = -0.1.$$

To predict the motion of the pendulum system over time, these final values are used to approximate the next iteration. 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.

B. 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 Euler method, then anjusted 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 Euler method is applied to predict v and y at time $t_n + 1$:

$$v_{\text{prediction}} = v_n + h \times -\omega^2 x_n$$
$$x_{\text{prediction}} = x_n + h \times v_n. \qquad (7)$$

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

$$v_\text{corrected} = v_\text{prediction}. \tag{8}$$

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

$$v_\text{corrected} = v_n + h\left(\frac{3}{2}(-\omega^2 x_n) - \frac{1}{2}(-\omega^2 x_{n-1})\right). \tag{9}$$

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

$$x_\text{corrected} = x_n + \frac{h}{2}(v_n + v_\text{corrected}). \tag{10}$$

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

$$\begin{aligned} x_{n+1} &= x_\text{corrected} \\ v_{n+1} &= v_\text{corrected}. \end{aligned} \tag{11}$$

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

$$\begin{aligned} v_\text{prediction} &= 0 + 0.01 \times (-10 \times 1) = -0.1 \\ x_\text{prediction} &= 1 + 0.01 \times 0 = 1, \end{aligned} \tag{12}$$

followed by the corrector,

$$\begin{aligned} v_{n+1} &= -0.1 \quad \text{(since it is the first step)} \\ x_{n+1} &= 1 + \frac{0.01}{2}(0 - 0.1) = 0.9995. \end{aligned} \tag{13}$$

As with RK2, to predict the motion of the pendulum system over time, these final values are used to approximate the next iteration. The P-C method involves 4 main calculations per iteration for 2nd order ODEs.

C. Backward Euler Method

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

$$\begin{aligned} x_{n+1} &= x_n + h \times v_{n+1} \\ v_{n+1} &= v_n + h \times (-\omega^2 x_{n+1}). \end{aligned} \tag{14}$$

The system of equations is then solved. Although the system of equations is simple in this example, the potential nonlinearity of equations in more complex systems requires the use of numeric solvers, as will be implemented in the test scripts.

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

$$\begin{aligned} x_{n+1} &= 1 + 0.01 \times v_{n+1} \\ v_{n+1} &= 0 - 0.01 \times 10 \times x_{n+1}. \end{aligned} \tag{15}$$

Using the numeric solver `fsolve` from the `scipy.optimize` Python module yields the following results:

$$\begin{aligned} x_{n+1} &= 0.9990 \\ v_{n+1} &= -0.0999. \end{aligned} \tag{16}$$

Although this method only involves 2 fundamental calculations per iteration, there is no definite number of operations involved due to the numerical solving step which involves external libraries and its dependency on the complexity of the equations themselves.

III.   Complexity Optimization

The complexity of each numerical method can be quantified by measuring their runtimes. Each method was applied to approximate three practical equations over 10 trials: The equation of motion of a simple harmonic oscillator,

$$\frac{\mathrm{d}^2 x}{\mathrm{d}t^2} = -\omega^2 x, \tag{17}$$

the equation of motion of a damped harmonic oscillator,

$$\frac{\mathrm{d}^2 x}{\mathrm{d}t^2} + b\frac{\mathrm{d}x}{\mathrm{d}t} + kx = 0, \tag{18}$$

and the Van der Pol oscillator equation,

$$\frac{\mathrm{d}^2 x}{\mathrm{d}t^2} - \mu(1 - x^2)\frac{\mathrm{d}x}{\mathrm{d}t} + x = 0. \tag{19}$$

The following data

IV.   Accuracy Optimization

V.   Evaluation

A. Second Subheading

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do. Inline equation $p = \frac{1}{2}\rho v^2$ followed by separate equation Equation (20).

$$\sqrt{a^2 + b^2} = c \tag{20}$$

1. Example Figures and Tables

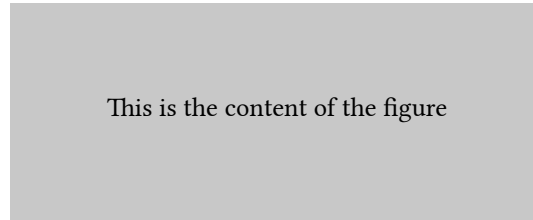Here's an example figure, Fig. 1:

Fig. 1 This is the caption of the figure.

Likewise, here's an example table, Table 1:

Table 1 Here's an example table.

	Area	Parameters
Cylinder	$\pi h\dfrac{D^2 - d^2}{4}$ (21)	h: height D: outer radius d: inner radius
Tetrahedron	$\dfrac{\sqrt{2}}{12}a^3$ (22)	a: edge length

Finally, some concluding text under the table.

2. Tertiary Heading Level

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore.

3. Continuing Tertiary Heading

- List of items
- List of items, item 2
- Etc.

B. Big 2

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus.

VI.   Related Work

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut.

References

[1]   Adams, D., "The Hitchhiker's Guide to the Galaxy", Pan Books, 1979.

[2]   Horowitz, H., and Hill, W., "The Art of Electronics", Cambridge University Press, Cambridge, England, 2015.