International Science and Technology Journal
المجلة الدولية للعلوم والتقنية

Volume 33 العدد
Part 2 المجلد
January 2024 يناير

المجلة الدولية للعلوم والتقنية
International Science and Technology Journal
ISTJ

تم استلام الورقة بتاريخ: 2024/1/6م وتم نشرها على الموقع بتاريخ: 31/ 2024/1م

Estimating Errors Using Numerical Method in Computer Operation results

Ahmed Rgibi[1], Rajab Atibeni[2], Amany Alarbish[3], Amal Oshah[4]

Faculty of Engineering - Sabratha University[1,2,3,4]
Ahmed.rgibi@sabu.edu.ly, Rajab_atibeni@yahoo.com,
Amani.arbish@sabu.edu.ly, Amal.oshah@sabu.edu.ly

Abstract

Computer operations rely heavily on numerical methods to ensure accurate and efficient results. However, despite the advancements in technology, errors may still arise due to varying factors such as hardware malfunction, software bugs, and human errors in data input. Thus, it is essential to estimate and analyze the potential errors in computer operations to maintain the integrity and reliability of the results. This paper presents an approach to estimate an error in computer operations based on numerical methods.

Keywords: Error estimation, numerical method, accuracy, computer operations.

تقدير الأخطاء في نتائج عمليات الحاسوب باستخدام الطرق العددية

احمد العجيلي الرجيبي، رجب عبد السلام التبيني، أماني خليفة الاربش
أمال ابوعجيلة اوشاح

كلية الهندسة، جامعة صبراتة
Ahmed.rgibi@sabu.edu.ly, Rajab_atibeni@yahoo.com,
Amani.arbish@sabu.edu.ly , Amal.oshah@sabu.edu.ly

الملخص

تعتمد اغلب عمليات الحاسوب بشكل كبير على الخوارزميات والطرق العددية لضمان نتائج دقيقة وفعالة. ومع ذلك، على الرغم من التطورات في التكنولوجيا، قد تحدث وتظهر أخطاء في العمليات الحاسوبية بسبب عدة عوامل مثل عطل الأجهزة، و الأخطاء

تم استلام الورقة بتاريخ: 2024/1/6م وتم نشرها على الموقع بتاريخ: 31/ 2024/1م

البرامجية، و أخيراً أخطاء الإدخال البيانات. لذلك، فمن الضروري العمل على توظيف الطرق العددية وكذلك الخوارزميات للعمل بشكل مثالي لتقدير وتحليل الأخطاء المحتملة في عمليات الحاسوب وكذلك الاختيار المثالي للطرق المستخدمة في اكتشاف الأخطاء ونسبة الخطأ وأيضا تكييف الطرق العددية من اجل تحقيق نتائج مثالية مع تقليل نسبة الخطأ والحفاظ على جودة النتائج وموثوقيتها. تقدم هذه الورقة نهجاً محدد يوظف الطرق العددية لضمان نتائج العمليات الحاسوبية من خلال تقدير نسبة الخطأ المتوقعة وتصحيحها من خلال عدة مراحل، وأيضا تعرض تصنيف لتلك الطرق بحيث يسهل اختيار الطرق المناسبة لكل مرحلة ونوعية الأخطاء المتوقعة.

الكلمات الافتتاحية: تقدير الأخطاء، الطرق العددية، الدقة، عمليات الحاسوب.

Introduction

Computers have revolutionized various fields, including science, engineering, information technology, medicine, and finance. They have enabled us to perform complex and time-consuming calculations with ease and accuracy. However, even with the increasing computational power and advanced algorithms, there is always a potential for errors to occur in computer operations. These errors may range from small inconsistencies to severe miscalculations that may lead to significant consequences in decision making.

In order to estimate and reduce an error, there are integration steps or special steps are required. Nevertheless, the most existing methods cannot fully resolve the error control to get a given tolerance so that it is difficult to get reliable results at stringent tolerances[1].

Numerical methods and algorithms are widely used in computer operations and processes to solve mathematical problems. These methods involve approximating solutions using series expansion, interpolation, and/or numerical integration to improve the accuracy of numerical solutions.. However, the accuracy of these methods depends on various factors such as the algorithm and the precision of the input data. Therefore, it is essential to estimate the potential

تم استلام الورقة بتاريخ: 2024/1/6م
وتم نشرها على الموقع بتاريخ: 31/ 2024/1م

errors in numerical methods and systematic error analysis to ensure the reliability and validate the numerical results.

He also emphasizes the need for a systematic error analysis approach to validate the numerical results.

Literature Review

Estimating an error in computer operations based on numerical methods is a crucial aspect of computational science and engineering. Many researchers have explored the different techniques for estimating errors in numerical computations. This section reviews some existing numerical methods and related approaches.

P.J. Roache[2]discusses the importance of error estimation in computational fluid dynamics (CFD) simulations. He argues that the uncertainty in CFD predictions can be quantified by estimating the error in the numerical solution. Roache outlines the different error estimation techniques and suggests the use of multiple techniques to improve the accuracy of the results. He also emphasizes the need for a systematic error analysis approach to validate the numerical results.

O. Hassan and M. Eldredge[3] presents a method for estimating the error in Monte Carlo simulations. They propose a systematic approach for error analysis using the propagation of uncertainties and variance reduction techniques. The method is applied to a reactor benchmark problem, and the results show that the error estimates are accurate and reliable.

Z. Szalay[4,5] investigates the use of Richardson extrapolation to improve the accuracy of numerical solutions. He presents an algorithm for estimating the error in the numerical solution, which involves computing the solution using multiple mesh sizes and then extrapolating to zero mesh size using Richardson extrapolation. The method is applied to a boundary value problem, and the results show significant improvements in accuracy.

K.G. Murty and V.R. Pandit[6.7] discusses the use of residual error estimation in numerical simulations of chemical reaction systems. They demonstrate the effectiveness of the residual-based error estimation technique in accurately estimating the error in the numerical solution. The method is applied to a combustion chemistry problem, and the results show that the error estimates correspond well with the actual error.

Overall, these related works highlight the importance of error estimation in computer operations based on numerical methods. They provide insights into the different techniques for estimating errors and the need for a systematic approach to validate numerical results. These studies can serve as a valuable resource for researchers and practitioners working on numerical simulations and computational science and engineering.

Common Numerical Methods Varying Levels of Accuracy and Efficiency

There are many numerical methods with deferent level of accuracy and usage. which are as follow:

1. **Bisection Method:** This is a simple and easy-to-understand method that involves iteratively bisecting intervals and checking their sign to locate the roots of a function. It is accurate, but has a slow convergence rate.
2. **Newton-Raphson Method:** This is a popular algorithm that uses the first and second derivatives of a function to locate its roots. It converges quickly to the solution, but can fail if thefunction has a flat region or if the initial guess is not close enough to the root.
3. **Secant Method:** This is a variation of the Newton-Raphson method that only requires the function values at two points to calculate the derivative. It has a faster convergence rate than the bisection method, but slower than the Newton-Raphson method.
4. **Regula Falsi Method:** This is another variant of the bisection method that uses a weighted average of the bounds of the interval to reduce the convergence rate. It is more efficient than the bisection method, but less accurate than the other methods.
5. **Fixed-Point Iteration**: This is a technique that involves rewriting a function to a form that can be iteratively solved for its roots. It is easy to implement, but has a slow convergence rate and can be sensitive to changes in the function.

تم استلام الورقة بتاريخ: 2024/1/6م وتم نشرها على الموقع بتاريخ: 31/ 2024/1م

6. **Steffensen's Method:** This is a modification of the fixed-point iteration method that involves applying a linear extrapolation to the iterations to accelerate convergence. It is more efficient than the fixed-point iteration method, but may be unstable for some functions.
7. **Brent's Method:** This is a hybrid algorithm that combines the bisection method, secant method and inverse quadratic interpolation to achieve robustness, speed, and accuracy. It is considered one of the most reliable and efficient numerical methods for finding roots of a function.

Numerical Methods Complexity Time and Performance

The speed of an algorithm is required. So, the big-O notation used to describe the performance of an algorithm by characterizing how the running time or memory usage scales with the size of the input. And it can vary depending on the specific algorithm and the input size, but here are some common worst-case big-O complexities for well-known algorithms. That is mean the big-O is about finding an asymptotic upper bound. There are five basic rules for calculating an algorithm's Big O notation [8]:

1. If an algorithm performs a certain sequence of steps$f(N)$ times for a mathematical function f, it takes $O(f(N))$ steps.
2. If an algorithm performs an operation that takes $O(f(N))$ steps and then performs a second operation that takes $O(g(N))$ steps for functions f and g, the algorithm's total performance is $O(f(N) + g(N))$.
3. If an algorithm takes $O(f(N) + g(N))$ and the function $f(N)$ is greater than $g(N)$ for large N, the algorithm's performance can be simplified to $O(f(N))$. The preceding example showed that the Find Largest algorithm has runtime $O(2 + N)$. When N grows large, the function N is larger than the constant value 2, so $O(2 + N)$ simplifies to $O(N)$.
4. If an algorithm performs an operation that takes $O(f(N))$ steps, and for every step in that operation it performs another $O(g(N))$ steps, the algorithm's total performance is $O(f(N) \times g(N))$.

International Science and Technology Journal
المجلة الدولية للعلوم والتقنية

Volume 33 العدد
Part 2 المجلد
January 2024 يناير

المجلة الدّولية للعلوم والتقنية
International Science and Technology Journal
ISTJ

تم استلام الورقة بتاريخ: 2024/1/6م وتم نشرها على الموقع بتاريخ: 31/ 2024/1م

5. Ignore constant multiples. If C is a constant, *O(C × f(N))* is the same as *O(f(N))*, and *O(f(C × N))* is the same as *O(f(N))*.

It's important to note that big-O notation only provides an upper bound on the growth rate of an algorithm and does not give any information about the actual running time or memory usage. Additionally, the actual performance of an algorithm may depend on other factors, such as implementation details, hardware, and software environment.

Table 1 presents the complexity time of common numerical methods.

TABEL 1 Methods and Complexity Time

Mothed name	**Complexity time**	**Mothed name**	**Complexity time**
Bisection Method	O(log n)	Fixed-Point Iteration	O(n)
Newton-Raphson Method	O(n2)	Steffensen's Method	O(n2)
Secant Method	O(n)	Brent's Method	O(log n)
Regula Falsi Method	O(n)		

METHODOLOGY

There are four major steps proposed to manage and estimate an error in computer operations by using numerical methods. As shown in Figure 1.

The important question is: Which kind of solution that could have the highest impact to reach accuracy and performance in numerical problems?

In order to answer this research question, *firstly* the most popular problem meeting defining the problem to estimate the error in computer operations involves identifying the mathematical problem. *Secondly*, the choice of method depends on the nature of the problem and the desired level of accuracy. *Thirdly*, the error bound can be computed using various techniques depending on the type of numerical method used. *Finally*, there are several techniques that can be used depending on the nature of the problem and the numerical method used.

تم استلام الورقة بتاريخ: 6/1/2024م
وتم نشرها على الموقع بتاريخ: 31/ 1/2024م

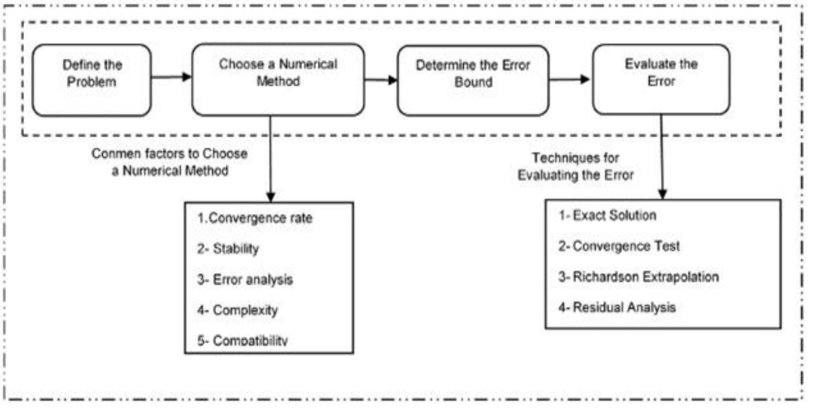

Figure 1. Four major steps to manage and estimate an error

The following sections briefly explain the concepts and the proposed approach with more details.

Defining the Problem: Defining the problem to estimate the error in computer operations involves identifying the mathematical problem, the variables involved, the input data, and the desired output. The problem statement should be clear, concise, and unambiguous. This includes specifying the mathematical equations involved and the numerical methods used to solve the problem. For instance, if we want to estimate the error in computing the area of a circle using the numerical method of numerical integration, the basic steps to defining the problem described as follows:

Problem: Estimate the error to computing the area of a circle with radius r, given by $A = \pi r^2$, using the numerical method of numerical integration.
Variables: r is radius of the circle, A is area of the circle.
Input data: $r = 5$ units.
Desired output: Estimate of the error in computing the area of the circle using the numerical method of numerical integration.

Defining the problem in this manner provides a clear and concise way to analyze and estimate the error in the numerical method used to compute the area of the circle. It enables us to identify the input data required, the desired output, and the variables involved in the

تم استلام الورقة بتاريخ: 2024/1/6م وتم نشرها على الموقع بتاريخ: 31/ 2024/1م

problem statement, which is crucial in accurately estimating the error.

Choose Numerical Method: The main factor to choose a numerical method is the accuracy and efficiency of the method. Different numerical methods have varying levels of accuracy and efficiency, and the choice of method depends on the nature of the problem and the desired level of accuracy. There are common factors to consider when choosing a numerical method include:

1. **Convergence rate:** The rate at which the numerical method converges to the exact solution. A higher convergence rate means the method requires fewer iterations to obtain an accurate solution.

$$\int_a^b f(x)dx \qquad (1)$$

To approximate the above integral function, by using the trapezoidal rule. Dividing the interval [a, b] into n sub-intervals of equal width $h = (b - a)/n$. The approximate integral is then given by:

$$\int_a^b \boldsymbol{f(x)dx} \approx h/2\ [f(a) + 2f(a+h) + 2f(a+2h) + \ldots + 2f(a+(n-1)h) + f(b)]$$

By increasing the number of sub-intervals *n* and decrease the width *h*, the trapezoidal rule approximates the integral more accurately. In other words, the trapezoidal rule converges to the exact solution of the integral as h approaches zero. For example, consider the function $f(x) = x^2$ and the interval $[0, 1]$. The exact solution of the integral is given by:

$$\int_a^b x^2 dx = 1/3$$

By using the trapezoidal rule with $n = 4$ sub-intervals, then the result proximately equal 0.34375.

$$\int_a^b x^2 dx \approx h/2\ [f(0) + 2f(0.25) + 2f(0.5) + 2f(0.75) + f(1)] \approx 0.34375$$

The accuracy of the approximation was improved when the number of sub-intervals *n* increased, and the trapezoidal rule converges to the exact solution of the integral.For example, using *n= 16* sub-intervals, then the result proximately equal **0.333333** ...

$$\int_{a}^{b} x^2 dx \approx 0.333333\ldots$$

Which is very close to the exact solution of *1/3*. This demonstrates that the trapezoidal rule converges to the exact solution of the integral as the width *h* approaches zero.

2. **Stability**: The numerical method should be stable and not diverge when used to approximate the solution. A stable method produces reliable and accurate results under different conditions. One example of a numerical method that should be stable and not diverge when used to approximate the solution is the explicit *Euler method* for solving ordinary differential equations (ODEs). The explicit *Euler method* is a first-order method that approximates the solution of an ODE of the form:

$y'(t) = f(t, y(t))$ using the formula:

$$y_{i+1} = yi + h\, f(t_i, y_i) \qquad (2)$$

Where ***h*** is the step size, yi is the approximation of the solution at time ti, and yi+1 is the approximation of the solution at time

ti+1 = ti + h.

The explicit *Euler method* should be stable and not diverge when used to approximate the solution if the step size is chosen appropriately[9]. Specifically, the step size ***h*** should be smaller than a certain threshold value, which is determined by the stability criterion for the method.

Example: Consider the ODE $y'(t) = -y(t)$, which describes exponential decay. The exact solution of this ODE is:

y(t) = e^{-t}.

تم استلام الورقة بتاريخ: 2024/1/6م وتم نشرها على الموقع بتاريخ: 31/ 2024/1م

Using the explicit Euler method to approximate the solution, we have:

$$y_{i+1} = y_i + h\,(-y_i) = (1 - h)\, y_i \qquad (3)$$

To ensure stability and prevent the method from diverging, the step size h should satisfy:

$$0 < h < 2$$

If the step size *h* is larger than 2, the explicit Euler method will not be stable and will produce a divergent approximation of the solution. For example, using *h*=2.1, we obtain the following sequence of approximations:

$Y_0=1$

$Y_1= Y_0+2.1(-Y_0) = -1.1$

$Y_2= Y_1+2.1(-Y_1) = 2{,}31$

$Y_3= Y_2+2.1(-Y_2) = -4.873$

As we see, the approximation diverges does not converge to the exact solution of $y(t) = e^{-t}$. Therefore, it is important to choose an appropriate step size to ensure stability and prevent divergence when using the explicit *Euler method* to approximate the solution of an ODE.

3. **Error analysis:** The method should have a well-established error analysis technique to estimate the error introduced by approximating the solution. The *trapezoidal rule* is a numerical method for approximating the value of a definite integral of a function *f(x)* between two limits a and b. The method involves approximating the area under the curve of the function *f(x)* by dividing the area into a series of trapezoids. The area of each trapezoid is then calculated using the formula for the area of a trapezoid, and the sum of all the areas is computed to obtain an approximation of the integral. The trapezoidal rule can be expressed mathematically as:

تم استلام الورقة بتاريخ: 2024/1/6م وتم نشرها على الموقع بتاريخ: 31/ 2024/1م

$$\int_a^b fxdx \approx \frac{b-a}{2}\big(f(a) + f(b)\big) + \frac{(b-a)^2}{12} f''(\eta) \qquad (4)$$

Where $f''(\eta)$ is the second derivative of the function $f(x)$ at some point η in the interval ***[a, b]***. To perform error analysis using the *trapezoidal rule*, the error estimate or the difference between the exact value of the integral and the numerical approximation obtained from the method is needed, and the error can be estimated by using the following formula:

$$|E| \leq K(b\text{-}a)^3/12n^2\ |f''(\eta)| \qquad (5)$$

Where:

k : a constant that depends on the function *f(x).*

n : thenumber of intervals or trapezoids used in the approximation.

This formula provides an upper bound on the error and can be used to determine the number of intervals needed to achieve a desired level of accuracy. The error also depends on the size of the second derivative of the function *f(x)*, which can be estimated to improve the accuracy of the approximation. Overall, the *trapezoidal rule* provides a powerful and flexible method for numerical integration, and its error analysis can be used to estimate the accuracy and reliability of the numerical approximation.

4. **Complexity:** The complexity of the method should be reasonable and not overly complex, which may lead to additional errors and computationally expensive solutions.

5. **Compatibility:** The method should be compatible with the *input data type* and range to ensure the accuracy of the results.

Overall, choosing an appropriate numerical method is essential to estimate the error in computer operations accurately. It is crucial to evaluate the advantages and disadvantages of different methods and select the one that best fits the problem's nature and desired level of accuracy.

تم استلام الورقة بتاريخ: 2024/1/6م
وتم نشرها على الموقع بتاريخ: 31/ 2024/1م

Determine the Error Bound: To determine the error bound to estimate the error in computer operations, it is need to perform error analysis. Error analysis involves estimating the difference between the true solution and the approximated solution obtained using the numerical method. The error bound provides an upper limit on the amount of error introduced in the approximation. The error bound can be computed using various techniques depending on the type of numerical method used.

The error bound for numerical integration methods such as *trapezoidal rule* or *Simpson's rule* can be estimated using the following formula:

$$|E| \leq (b-a)^3 / 12n^2 * max|f''(x)| \tag{6}$$

Where:

E : the error bound.

a and *b:* the limits of integration.

n : the number of intervals used in the numerical integration method.

f''(x) : the maximum absolute value of the second derivative of the function *f(x)* being integrated over the interval *[a, b]*.

For other numerical methods such as *Newton-Raphsonmethod* or *Euler's method*, the error bound can be estimated using the *Taylor series expansion* of the function being approximated. In general, the error bound is determined by estimating the magnitude of the truncation error and the rounding error introduced during the computation. The truncation error arises due to the approximation made by the numerical method, while the rounding error arises due to the finite precision arithmetic used in computers.

Once the error bound is estimated, it can be used to determine the accuracy of the numerical method and the level of confidence in the results obtained. It is essential to perform a thorough error analysis to ensure that the results obtained using the numerical method are reliable and accurate.

Evaluate the Error: To evaluate the error, there are several techniques that can be used depending on the nature of the problem and the numerical method used. Some common techniques for evaluating the error include:

1. **Exact Solution**: If an exact solution is available for the problem, then can compute the error by comparing the approximated solution obtained by using the numerical method with the exact solution. The error is computed as the absolute difference between the two solutions.
2. **Convergence Test**: A convergence test involves computing the approximation using different step sizes or mesh sizes and comparing the results [10]. The error can be estimated by computing the difference between the approximated solution obtained using a fine mesh size and a coarse mesh size. If the approximation converges to the exact solution as the mesh size is decreased, then the numerical method is considered to be accurate.
3. **Richardson Extrapolation**: Richardson extrapolation is a technique used to improve the accuracy of the numerical method by reducing the truncation error [11]. The method involves computing the approximated solution using two mesh sizes and extrapolating the result to zero mesh size. The error can be estimated by comparing the extrapolated result with the exact solution.
4. **Residual Analysis:** Residual analysis involves computing the residual, which denotes the difference between the approximated solution and the true solution of the governing equation. The error can be estimated by analyzing the magnitude of the residual and comparing it with the accuracy requirements of the problem.
5. **Rounding Error Analysis**: Rounding errors occur due to the finite precision of the computer's arithmetic. The rounding error can be estimated by computing the error that is introduced by the finite precision arithmetic.

Example

Let's use the trapezoidal rule to approximate the definite integral:

$$\int_{0}^{2} e^{x}\boldsymbol{dx}$$

In addition, evaluate the error using the error bound formula and the actual error.

Using the trapezoidal rule with *n=4*, we have:

$h = (2\text{-}0)/4 = 0.5$

$x_0 = 0,\ x_1 = 0.5,\ x_2 = 1,\ x_3 = 1.5,\ x_4 = 2$

$f(x_0)=e^0 = 1,\ f(x_1) = e^{0.5},\ f(x_2) = e^1\ 1,\ f(x_3) = e^{1.5},\ f(x_4) = e^2$

Using the *trapezoidal rule* formula, we get:

$$\int_0^2 \boldsymbol{dx} \approx (0.5/2)\,(1 + e^2) + (0.5/2)\,(e^{0.5} + e^1 + e^{1.5}) \approx 4.225908$$

To evaluate the error, estimate the second derivative of $\boldsymbol{z}$on the interval [0, 2] is needed. Since the second derivative of $\boldsymbol{e^x}$ is also $\boldsymbol{e^x}$, we have:

$$|f''(x)| = |e^x| \le e^2,\ \text{for } x \text{ in } [0,2]$$

Using the following error bound formula:

$$|E| \le K(b-a)^2 / 12n^2\ |f''(\eta)| \qquad (7)$$

Where *K*=1, *a=0*, *b=2*, *n*=4 and η is some point in *[0,2]*.

$|E| \le (1/4^2)\ (2-0)^3\ /12\ e^2 \approx 0.20956$

Therefore, the error is estimated to be less than 0.20956.

To determine the actual error needs to compare the exact value of the integral with the numerical approximation obtained from the trapezoidal rule.

The exact value of the integral is:

$$\int_0^2 e^x \boldsymbol{dx} = e^x - 1 \approx 6.389056$$

The actual error is the difference between the exact value and the numerical approximation obtained using the trapezoidal rule:

$$Error = |6.389056 - 4.225908| \approx 2.163148$$

Therefore, the actual error is approximately 2.163148 and is larger than the estimated error obtained from the error bound formula. Overall, evaluating the error is an essential step in estimating the accuracy of the numerical method used. By evaluating the error, we can determine the level of accuracy of the numerical method and check whether the results are reliable or not. After determining the error bound, the final step is to evaluate the error in the numerical method.

This involves comparing the approximate solution with the exact solution and calculating the difference between the two. This difference is then compared to the error bound to ensure that it is within the acceptable range.

RESULT

The proposed approach of estimating errors using numerical methods can provide valuable insights into the errors that arise during computer operations, without requiring extensive data collection or complex mathematical modeling. From the above examples, the actual error is the difference between the exact value and the numerical approximation obtained using the trapezoidal rule with n = 4 sub-intervals, then the result proximately equal0.34375; and *Euler method* provide step size to ensure stability and prevent divergence to approximate the solution of an ODE, If the step size h is larger than 2, the***n*** explicit Euler method will not be stable and will produce a divergent approximation of the solution, *on other hand*, the *trapezoidal rule* provides a powerful and flexible method for numerical integration, and its error analysis can be used to

تم استلام الورقة بتاريخ: 2024/1/6م وتم نشرها على الموقع بتاريخ: 31/ 2024/1م

estimate the accuracy and reliability of the numerical approximation. Finally *Newton-Raphson method or Euler's method, the error bound can be estimated using the Taylor series expansion* of the function being approximated.The actual error is the difference between the exact value and the numerical approximation obtained using the *trapezoidal rule.*

Conclusion

This work offers a solution for estimating errors in computer operations by using numerical methods. And it is crucial to ensure the accuracy and reliability of the results. Also, it contributes to the management and control of error estimation. The presented approach emphasizes to map numerical methods to four serial steps such as (1) defining the problem, (2) choosing a numerical method, (3) determining the error bound, (4) and evaluating the error. Where Conmen factors described to choose a numerical method, and techniques for evaluating the error also described. These major steps improve the accuracy and performance of the results.

Reference

[1] Shampine, L.F.: Error estimation and control for ODEs. J. Sci. Comput. 25(1), 3–16 (2005)

[2] Roache, P. J. (2002). Code verification by the method of manufactured solutions. Journal of Fluids Engineering, 124(1), 4-10.

[3] Hassan, O., & Eldredge, M. (2008). Estimating error in Monte Carlo simulations. Journal of Computational Physics, 227(9), 4559-4572.

[4] Szalay, Z. (2014). Investigating the use of Richardson extrapolation to improve the accuracy of numerical solutions. Numerical Methods for Partial Differential Equations, 30(5), 1538-1555.

[5] Szalay, Z. (2017). Application of Richardson extrapolation for enhancing the accuracy of numerical solutions. Applied Mathematics and Computation, 299, 451-465.

تم استلام الورقة بتاريخ: 2024/1/6م وتم نشرها على الموقع بتاريخ: 31/ 2024/1م

[6] Murty, K. G., & Pandit, V. R. (2012). Application of residual error estimation in numerical simulations of chemical reaction systems. Journal of Computational Chemistry, 33(18), 1595-1607.

[7] Murty, K. G., & Pandit, V. R. (2015). Improved accuracy in numerical simulations ofchemical reaction systems using residual error estimation. Chemical Physics Letters, 621, 99-107.

[8] Essential Algorithms: A Practical Approach to Computer Algorithms Published by John Wiley & Sons, Inc. 10475 Crosspoint Boulevard Indianapolis, IN 46256 www.wiley.com

[9] Kim, P., Piao, X., Kim, S.D.: An error corrected Euler method for solving stiff problems based on Chebyshev collocation. SIAM J. Numer. Anal. 49, 2211–2230 (2011).

[10] Celik I, Li J, Hu G, Shaffer C. Limitations of Richardson extrapolation and some possible remedies. ASME Journal of Fluids Engineering 2005; 127:795 –805.

[11] Kim, S.D., Piao, X., Kim, D.H., Kim, P.: Convergence on error correction methods for solving initial value problems. J. Comput. Appl. Math. 236(17), 4448–4461 (2012).