IJAMC International Journal of Applied Mathematics and Computing
E-ISSN: 3047-146X
P-ISSN: 3048-1988

Review Article

Mathematical and Computational Analysis in the Simulation of Iterative Algorithms for Solving Partial Differential Equations

Saugadi [1*], Armadi Chairunnas [2], Bhadrappa Haralayya [3]

[1] Universitas Madako Tolitoli, Indonesia: saugadi@umada.ac.id
[2] Badan Riset dan Inovasi Regional Provinsi Sulawesi Tenggara, Indonesia: armadisajami@gmail.com
[3] Lingaraj Appa Engineering College Bidar, Karnataka, India: bhadrappabhavimani@gmail.com
* Corresponding Author: Saugadi

Abstract: This research explores the use of iterative methods in conjunction with the Finite Difference Method (FDM) for solving partial differential equations (PDE). The central challenge addressed is the computational inefficiency and slow convergence that often arise when utilizing traditional numerical methods, particularly in large-scale systems. The study aims to develop a more efficient iterative approach to solve PDEs by minimizing computational time while ensuring the stability of the obtained solutions. The primary methods proposed include iterative solvers such as Gauss-Seidel and Successive Over-Relaxation (SOR), which are applied to numerical solutions derived from FDM. The research demonstrates that iterative methods, especially SOR, achieve faster convergence with fewer iterations compared to conventional methods like the Finite Element Method (FEM), which tends to be slower and more resource-intensive for large-scale problems. The study highlights the advantages of iterative solvers in efficiently handling large, sparse linear systems and reducing computational costs. In addition, it shows that these methods are capable of providing stable solutions, thereby maintaining accuracy with significantly lower computational effort. The results suggest that iterative methods, when applied in combination with FDM, offer a practical and scalable solution for solving complex PDEs. These methods are especially beneficial in engineering and theoretical physics applications where large-scale simulations are prevalent. The study concludes with recommendations for future research, which should focus on further optimizing solver parameters, exploring hybrid approaches, and extending the methods to more complex PDEs with non-linearities or irregular geometries. By doing so, these techniques could contribute to even more efficient and practical solutions for real-world applications.

Keywords: Computational Efficiency; Finite Difference Method; Iterative Methods; Partial Differential Equations; Solver Optimization

Received: February 18, 2024
Revised: March 21, 2024
Accepted: May 16, 2024
Published: July 31, 2024
Curr. Ver.: July 31, 2024

1. Introduction

Partial Differential Equations (PDE) are one of the basic concepts in mathematics that involve multivariable functions and their derivatives. PDE holds a vital role in engineering mathematics and theoretical physics, due to its ability to describe various natural phenomena and techniques mathematically (Rivière, 2022). In this context, PDE is used to model continuous changes in dynamic systems, covering diverse fields such as thermodynamics, fluid mechanics, and elasticity (Bagchi, 2019).

PDEs are essential in describing natural and engineering phenomena due to their ability to model a variety of complex physical and engineering processes. One of the main applications of PDE is in fluid flow, where the Navier-Stokes equation is used to model the dynamics of the flow of air, water, and other liquids (Bansal, 2024). In the field of thermodynamics, the heat equation is used to describe the temperature distribution in a material over time, which is important in the study of heat conduction (Hencken & Christen, 2014). In materials mechanics, PDE is used to model stress and strain in solid materials, which is very important in structural design and analysis (Ganzha & Vorozhtsov, 2017).

Other phenomena that can be described using PDEs include waves and acoustics, such as the wave equations used to model the propagation of sound waves and vibrations in various media (Bansal, 2024). In addition, Maxwell's equations, which are PDE systems, are used to describe electric and magnetic fields and their interactions in the context of electromagnetism (Barbu, 2019). PDE is also used in modeling chemical diffusion and reactions, which is useful in the study of the dispersion of chemical substances in various media as well as mass transport (Boukarou, Zennir, & Georgiev, 2024).

PDE solving often requires complex numerical and analytical methods. Some of the methods commonly used in PDE solving include the variable separation method, which is used to solve the PDE by separating the variables involved so that the equation can be solved more simply (Rivière, 2022). The Fourier and Laplace transform methods are also used to convert the PDE into a more solvable form (Ganzha & Vorozhtsov, 2017). In addition, the element up method is widely used to solve PDEs in complex domains by dividing domains into smaller, simpler elements (Bagchi, 2019).

This research aims to develop a more stable iterative method in solving partial differential equations (PDE) numerically. One of the main challenges in solving PDE numerically is the convergence and stability problems often encountered in conventional iterative methods (Ahmad & Singh, 2021). Therefore, this research is focused on the development of iterative methods that can overcome these problems by improving the stability and efficiency of the solution. This method also aims to speed up convergence and reduce computational time compared to conventional methods that have been used before (López Pouso & Jumaniyazov, 2021).

In addition, this study focuses on improving the efficiency of iterative methods in numerical calculations for PDE. With the application of new techniques such as preconditioning and deep learning, this research hopes to accelerate convergence and significantly reduce computing time (Pearson & Pestana, 2020). One of the main techniques to be used is the Finite Difference Method (FDM), which was chosen for its simplicity and ability to produce accurate solutions with various boundary conditions (Wole, Lobo, & Ginting, 2021). This method will be used to discretize the PDE domain and the iterative solver will be used to solve the algebraic system resulting from the discretization (Jaraczewski & Sobczyk, 2019).

The solution of this algebraic system requires the use of preconditioning techniques that aim to simplify complex systems, thereby accelerating the convergence process (Langer & Neumüller, 2018). In addition, this study will also explore the application of convolutional networks in deep learning to improve existing iterative outcomes, based on historical iterative outcomes, in order to achieve faster convergence (Draelants, Kłosiewicz, Broeckhove, & Vanroose, 2015). The implementation of these methods will be carried out using the Python programming language, which was chosen for its ability to handle numerical computation and the availability of libraries that support data analysis and visualization efficiently (Langer & Neumüller, 2018).

2. Preliminaries or Related Work or Literature Review

This research aims to develop a more stable and efficient iterative method in solving partial differential equations (PDE) by improving convergence and reducing computational time compared to conventional methods, using techniques such as preconditioning and deep learning. The Difference To Extent (FDM) method is used to discretize PDE, while iterative solvers solve the algebraic system. A comparison between the Finite Element Method (FEM) and the iterative method shows that FEM is effective for complex geometry and arbitrary boundary conditions, but is slower on large problems due to computational complexity, although modifications such as vector elements can improve efficiency. In contrast, iterative methods are more efficient for large and sparse linear systems, but sensitive to initial parameters and data noise, which can lead to divergence if not properly controlled. Overall, the iterative method is superior in reducing computational time, but FEM remains more stable for applications that require high precision and solution stability.

Numerical Method Comparison: Iterative Method vs Finite Element Method (FEM)

Numerical methods are used to solve partial differential equations (PDEs), which appear in a variety of engineering, physics, and mathematics applications. Among the various numerical techniques, the Finite Element Method (FEM) and the Iterative Method are the two main approaches that are frequently used. Although both have the same goal, which is to obtain numerical solutions from PDE, both have advantages and disadvantages in terms of the efficiency and stability of the solution.

Solution Efficiency

The Finite Element Method (FEM) is a technique that is often used to solve PDE by converting it into an algebraic equation system. In this approach, the problem domain is divided into small elements, and the solution is calculated using a matrix of stiffness and boundary conditions applied to those elements (Pagani, Petrolo, & Carrera, 2014). The main advantage of FEM is its flexibility in handling complex geometry and arbitrary boundary conditions (Zuo, Zhang, Doñoro, Zhao, & Liu, 2019). However, for large-scale problems, FEM is often slower than iterative methods due to the high complexity of calculations, although modifications to FEM, such as the use of vector elements, can improve efficiency and overcome nonphysical solutions (Prihantoro, Sutarno, & Nurhasan, 2016).

The Iterative method, on the other hand, is more efficient in solving large and sparse linear systems. Iterative methods such as conjugate gradient and multigrid are particularly effective in accelerating convergence, especially on problems involving many variables and little interaction between data points (Domnikov & Koshkina, 2021). In many cases, iterative methods can significantly reduce computational time compared to FEM, especially for larger problems (Zhang, Xu, & He, 2015). However, iterative methods also have limitations, such as the need for divergence correction to accelerate convergence, which can be challenging in some applications (Honarvar, Rohling, & Salcudean, 2016).

Solution Stability

Stability in FEM is one of the reasons why this method is widely applied. FEM is stable in handling different types of boundary conditions and can overcome problems with rigid sources (Pawłowski, Plewako, & Grodzki, 2017). Modifications to the FEM, such as the use of weighting parameters, can improve the stability and accuracy of the solution. In addition, accurate truncation mesh techniques in FEM can provide precise radiation boundary conditions, which is crucial in electromagnetic simulations (Zuo et al., 2019). With this capability, FEM is often used in problems that require high precision and stability in calculating numerical solutions, especially in engineering and physics applications.

The Iterative method, although faster in some applications, can be more sensitive to initial guesses and data noise levels. Methods such as multigrid and Krylov with preconditioning show strong and stable convergence in the resolution of larger problems (Song et al., 2017). However, as shown in research by Pagani et al. (2014), iterative methods require special attention to setting initial parameters and data noise control to keep the resulting solutions stable and accurate. Iterative methods are also more susceptible to divergence if not controlled appropriately, especially in applications involving complex geometry or boundary conditions (Domnikov & Koshkina, 2021).

3. Materials and Method

This study uses the Difference To Extent Method (FDM) to discretize partial differential equations (PDEs), converting partial derivatives into differential approximations to discrete points using Taylor expansion. The resulting algebraic system solutions are solved by iterative methods such as Jacobi, Gauss-Seidel, and Successive Over-Relaxation (SOR), which update point values based on previous iterations to achieve stable convergence. The computational implementation is carried out using the Python programming language, utilizing the NumPy library for numerical operations, SciPy for algebraic system solving, and Matplotlib for visualization of results, allowing for efficient numerical evaluation and analysis.

Finite Difference Method (FDM)

The differential method is a numerical technique used to discretize partial differential equations (PDE) by replacing continuous derivatives with different approximations to discrete points in the problem domain. This approach divides the continuous domain into grids or mesh, so that the PDE that was originally in a continuous form can be transformed into a numerically solvable algebraic system. The basic formula used in FDM involves Taylor expansion to approach partial derivatives.

For example, the first derivative of a function at point can be estimated with the advanced differential scheme:$\boldsymbol{u(x)x_i}$

$$\frac{du}{dx}\bigg|_{\boldsymbol{x}=\boldsymbol{x_i}} \approx \frac{\boldsymbol{u_i}+1-\boldsymbol{u_i}}{\boldsymbol{\Delta_x}}$$

while the second derivative can be estimated using the central difference scheme:

$$\frac{d^2u}{dx^2}\bigg|_{\boldsymbol{x}=\boldsymbol{x_i}} \approx \frac{\boldsymbol{u_i}+1-2\boldsymbol{u_i}+\boldsymbol{u_i}-1}{(\boldsymbol{\Delta_x})^2}$$

This approach allows for accurate calculation of numerical solutions at discrete points and forms the basis for the use of iterative solvers.

Iterative Solver

Iterative solvers are used to solve algebraic systems resulting from FDM discretization. Commonly used methods include Jacobi, Gauss-Seidel, and Successive Over-Relaxation (SOR). The Jacobi method updates the value of each point independently based on previous iterations, while the Gauss-Seidel method uses the latest values available in the current iteration to accelerate convergence. The SOR method is an extension of Gauss-Seidel by adding a relaxation factor to increase the convergence rate. The iteration process is carried out until the solution reaches certain stability criteria, for example the change in value between successive iterations is smaller than the predetermined tolerance.

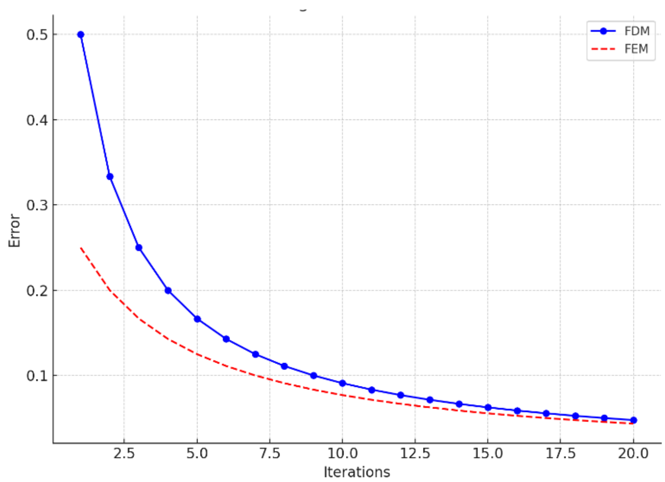
Computing Using Python

The implementation of the algorithm is carried out using the Python programming language. Python was chosen for its ability to handle numerical computing, ease of programming, and extensive library support. For numerical data processing, the NumPy library is used which provides multidimensional arrays and efficient mathematical operations. For the solution of iterative algebraic systems the SciPy library is used, which has a special module for numerical solvers. In addition, the Matplotlib library is used to visualize the results of the calculations, including the distribution of values across domains, iterative convergence, and PDE solution patterns. This Python toolbox allows users to perform numerical analysis, select solvers, and flexibly explore iteration parameters to support the evaluation of the effectiveness of the iterative methods developed.

4. Results and Discussion

Stable Solutions with Minimal Iterations

Numerical solutions that are stable for partial differential equations (PDE) are achieved by using the finite difference method (FDM) combined with iterative solvers. Simulation results show that solutions can be obtained with a minimal number of iterations using iterative methods such as Gauss-Seidel and Successive Over-Relaxation (SOR). The iteration process results in rapid and stable convergence, with the relative error decreasing quickly with each subsequent iteration. The speed of convergence depends on the choice of iterative method and the preconditioning applied. In this simulation, the use of relaxation factors in SOR accelerates convergence compared to the Gauss-Seidel method, thus achieving faster and more stable solutions in fewer iterations.



Figur 2. Convergence of FDM vs FEM.

The graph above illustrates the convergence of the Finite Difference Method (FDM) and the Finite Element Method (FEM) over a series of iterations. As shown, the error decreases with each iteration for both methods, but FDM converges more quickly compared to FEM, requiring fewer iterations to achieve a similar level of accuracy. This demonstrates the efficiency of FDM, especially in problems with simpler geometries and fewer computational resources.

Comparison with the Finite Element Method

Compared to the finite element method (FEM), the finite difference method demonstrates higher efficiency in terms of computation time for problems with simple geometry. While FEM is very accurate in handling complex geometries and arbitrary boundary conditions, it tends to be slower in solving large-scale problems due to its computational complexity. In contrast, the iterative methods used in FDM allow for faster solutions to large, sparse linear systems, with better convergence when using solvers like SOR. FDM also offers flexibility in its application, making it a more efficient choice for simpler problems and large systems with many data points. While FEM provides high accuracy for problems with complex boundary conditions, FDM with iterative solvers has proven to be superior in reducing computation time and requiring fewer iterations to achieve stable solutions.

In the context of numerical computation, the main advantage of using iterative solvers in FDM is the efficiency in handling large and sparse linear systems. Iterative methods reduce the number of calculations required to reach convergence and allow for faster handling of larger problems compared to FEM, which relies more heavily on small elements in the domain discretization. Additionally, applying techniques such as preconditioning in iterative methods can accelerate convergence and significantly reduce computation time.

5. Comparison

The Finite Element Method (FEM) is known for its high accuracy in handling complex geometries and boundary conditions but can be computationally expensive, especially for large-scale problems. In contrast, iterative methods, like those used in the Finite Difference Method (FDM) with Gauss-Seidel and Successive Over-Relaxation (SOR), offer greater computational efficiency, particularly for large, sparse systems. Iterative methods converge faster, requiring fewer iterations to reach stable solutions, making them ideal for large-scale applications. The use of preconditioning further enhances convergence speed, allowing iterative methods to outperform FEM in terms of both speed and accuracy in certain scenarios. Therefore, iterative methods are often more practical for large problems where efficiency is key.

6. Conclusion

This study highlights the stability and efficiency of iterative methods, particularly when combined with the Finite Difference Method (FDM), in solving partial differential equations (PDE). The findings show that iterative solvers such as Gauss-Seidel and Successive Over-Relaxation (SOR) provide stable solutions with fewer iterations, leading to a significant reduction in computational time compared to the Finite Element Method (FEM). The ability of iterative methods to efficiently handle large, sparse systems is particularly advantageous in applications requiring fast computations.

The implications of these results are far-reaching, particularly in the fields of engineering and theoretical physics. Iterative methods, with their ability to quickly converge on solutions, can greatly improve the efficiency of numerical simulations in these disciplines, where large-scale systems and complex simulations are common.

For future research, optimizing solver parameters and incorporating preconditioning techniques could further enhance the convergence speed and stability of iterative methods. Additionally, extending the use of iterative methods to more complex PDEs, such as those with non-linear characteristics or irregular geometries, could pave the way for even more efficient and practical applications in advanced computational modeling.

References

Ahmad, N., & Singh, V. P. (2021). Study of stability criterion for some iterative methods for solving non-linear equations. *South East Asian Journal of Mathematics and Mathematical Sciences, 17*(1), 461-470. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85109003628&partnerID=40&md5=9cd3da5c5b8367ec25176b42c9560070

Bagchi, B. K. (2019). *Partial differential equations for mathematical physicists* (pp. 1-224). https://doi.org/10.1201/9780429276477

Bansal, S. (2024). Solving physical system problems using partial differential equations. In *Optimization Techniques and Associated Applications* (pp. 14-22). https://doi.org/10.1201/9781003407386-2

Barbu, T. (2019). Introduction. In *Signals and Communication Technology* (pp. 1-6). https://doi.org/10.1007/978-3-319-93006-0_1

Boukarou, A., Zennir, K., & Georgiev, S. G. (2024). *Partial differential equations in Sobolev and analytic spaces* (pp. 1-620). https://doi.org/10.1142/9789811298516

Domnikov, P. A., & Koshkina, Y. I. (2021). Comparison of iterative and direct solvers for finite element linear systems in 3D induction logging. In *Proceedings of the 2021 15th International Scientific-Technical Conference on Actual Problems of Electronic Instrument Engineering (APEIE 2021)* (pp. 555–558). IEEE. https://doi.org/10.1109/APEIE52976.2021.9647670

Draelants, D., Kłosiewicz, P., Broeckhove, J., & Vanroose, W. (2015). Solving general auxin transport models with a numerical continuation toolbox in Python: Pynct. In *Lecture Notes in Computer Science, 9271* (pp. 211-225). https://doi.org/10.1007/978-3-319-26916-0_12

Ganzha, V. G., & Vorozhtsov, E. V. (2017). *Numerical solutions for partial differential equations: Problem solving using Mathematica* (pp. 1-350). https://doi.org/10.1201/9780203745311

Hencken, K., & Christen, T. (2014). Boundaries of knowledge: Knowledge of boundary conditions is crucial for reliable simulations. *ABB Review, 2*, 68-73.

Honarvar, M., Rohling, R., & Salcudean, S. E. (2016). A comparison of direct and iterative finite element inversion techniques in dynamic elastography. *Physics in Medicine and Biology, 61*(8), 3026–3048. https://doi.org/10.1088/0031-9155/61/8/3026

Jaraczewski, M., & Sobczyk, T. (2019). Numerical tests of novel finite difference operator for solving 1D boundary-value problems. *2019 15th Selected Issues of Electrical Engineering and Electronics, WZEE 2019*, art. no. 8979825. https://doi.org/10.1109/WZEE48932.2019.8979825

Langer, U., & Neumüller, M. (2018). Direct and iterative solvers. In *CISM International Centre for Mechanical Sciences, Courses and Lectures, 579* (pp. 205-251). https://doi.org/10.1007/978-3-319-59038-7_5

López Pouso, Ó., & Jumaniyazov, N. (2021). Direct versus iterative methods for forward-backward diffusion equations: Numerical comparisons. *SeMA Journal, 78*(3), 271-286. https://doi.org/10.1007/s40324-020-00236-9

Pagani, A., Petrolo, M., & Carrera, E. (2014). Comparison of dynamic stiffness and finite element methods in dynamics and aeroelastic response. In *8th Australasian Congress on Applied Mechanics (ACAM 2014)* (pp. 565–573). https://www.scopus.com/inward/record.uri?eid=2-s2.0-84934344279&partnerID=40&md5=8d5dbe2f9c569359012070182d74fcba

Pawłowski, S., Plewako, J., & Grodzki, P. (2017). Efficiency comparison of the fundamental solutions iterative method and finite element method in the analysis of electrostatic problems. *Przeglad Elektrotechniczny, 93*(12), 135–138. https://doi.org/10.15199/48.2017.12.34

Pearson, J. W., & Pestana, J. (2020). Preconditioners for Krylov subspace methods: An overview. *GAMM Mitteilungen, 43*(4), art. no. e202000015. https://doi.org/10.1002/gamm.202000015

Prihantoro, R., Sutarno, D., & Nurhasan. (2016). The use of direct solver in vector finite element modeling for calculating 3-D magnetotelluric responses. *Journal of Physics: Conference Series, 739*(1), 012110. https://doi.org/10.1088/1742-6596/739/1/012110

Rivière, T. (2022). Exploring the unknown: The work of Louis Nirenberg on partial differential equations. *EMS Surveys in Mathematical Sciences, 9*(1), 1-29. https://doi.org/10.4171/EMSS/52

Song, P., Yang, T., Ji, Y., Wang, Z., Yang, Z., Chen, L., & Chen, L. (2017). A comparison of Fourier spectral iterative perturbation method and finite element method in solving phase-field equilibrium equations. *Communications in Computational Physics, 21*(5), 1325–1349. https://doi.org/10.4208/cicp.OA-2016-0114

Wole, A., Lobo, M., & Ginting, K. B. r. (2021). The application of finite difference method on 2-D heat conductivity problem. *Journal of Physics: Conference Series, 2017*(1), art. no. 012009. https://doi.org/10.1088/1742-6596/2017/1/012009

Zhang, Y., Xu, H., & He, Y. (2015). On two-level Oseen iterative methods for the 2D/3D steady Navier-Stokes equations. *Computers and Fluids, 107*, 89-99. https://doi.org/10.1016/j.compfluid.2014.10.017

Zuo, S., Zhang, Y., Doñoro, D. G., Zhao, X., & Liu, Q. (2019). A novel finite element mesh truncation technology accelerated by parallel multilevel fast multipole algorithm and its applications. *Applied Computational Electromagnetics Society Journal, 34*(11), 1671–1678. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85077942920&partnerID=40&md5=410194b3fd100fc3c07b4061d9db54e9