On the Computationally Efficient Numerical Solution to the Helmholtz Equation

1 author:

Some of the authors of this publication are also working on these related projects:

Devising efficient algorithms for prediction, smoothing and filtering in the context of data assimilation. Besides, we do modelling as a side hustle.   View project

On the Computationally Efficient
Numerical Solution to the Helmholtz Equation

Angwenyi N. David, Lawi George, Ojiema Michael

Department of Mathematics
Masinde Muliro University of Science and Technology
P. O. Box 190–50100, Kakamega, Kenya

Owino Maurice

Department of Mathematics and Computer Science
University of Kabianga
P.O. Box 2030–20200, Kericho, Kenya

Abstract

Named after Hermann L. F. von Helmholtz (1821-1894), Helmholtz equation has obtained application in many fields: investigation of acaustic phenomena in aeronautics, electromagnetic application, migration in 3-D geophysical application, among many other areas. As shown in [2], Helmholtz equation is used in weather prediction at the Met Office in UK. Inefficiency, that is the bottleneck in Numerical Weather Prediction, arise partly from solving of the Helmholtz equation. This study investigates the computationally efficient iterative method for solving the Helmholtz equation. We begin by analysing the condition for stability of Jacobi Iterative method using Von Neumann method. Finally, we conclude that Bi-Conjugate Gradient Stabilised Method is the most computationally efficient method.

Keywords: Computationally Efficient, Helmholtz Equation

1. Introduction

In Numerical Weather Prediction, the discretisation of governing equations leads to the Helmholtz equation, which is solved for the increment in pressure field Π' [3]. The 2-dimensional Helmholtz equation is given by;

$$\nabla^2 \Pi'(x,y) + \omega^2 \Pi'(x,y) = \Phi(x,y) \tag{1.1}$$

where ∇^2 is the Laplacian in rectangular coordinates, ω is the wave number and Φ is the value of the pressure field calculated from the previous time step. Solving of 3-dimensional version of the Helmholtz equation alone takes about a third of the total forecast time [2]. As a result, optimal and efficient methods are being sought for solving this equation. In this study, a simple sample example of the Helmholtz problem is solved using Gauss-Seidel, Jacobi, Successive-Over-Relaxation, Conjugate Gradient, Bi-Conjugate Gradient, Bi-Conjugate Gradient Stabilised, Quasi-Minimal Residual and Generalised Minimal Residual Methods. Use is made of MATLAB for computation. Comparison of these methods is done in regards to computational efficiency. Before that, we analyse the stability of the 2-dimensional Helmholtz equation.

2. Discretisation of the Helmholtz equation

Let $\Phi(x,y) = 0$. This is to ease the analysis of stability. Therefore,

$$\nabla^2 \Pi'(x,y) + \omega^2 \Pi'(x,y) = 0 \tag{2.1}$$

Central difference approximation of (1) yields the following difference equation:

$$\frac{\Pi'_{i+1,j} - 2\Pi'_{i+,j} + \Pi'_{i-1,j}}{(\Delta x)^2} + \frac{\Pi'_{i,j+1} - 2\Pi'_{i,j} + \Pi'_{i,j-1}}{(\Delta y)^2} + \omega^2 \Pi'_{i,j} = 0 \tag{2.2}$$

To simplify further, we let $\Delta x = \Delta y = h$. This substitution and multiplication by h^2 leads to:

$$\Pi'_{i+1,j} - 2\Pi'_{i,j} + \Pi'_{i-1,j} + \Pi'_{i,j+1} - 2\Pi'_{i,j} + \Pi'_{i,j-1} + (h\omega)^2 \Pi'_{i,j} = 0 \tag{2.3}$$

Collecting like terms together in (2.3) leads to:

$$(4 - h^2\omega^2)\Pi'_{i,j} = \Pi'_{i+1,j} + \Pi'_{i-1,j} + \Pi'_{i,j+1} + \Pi'_{i,j-1} \tag{2.4}$$

Equation (2.4) is the discretised form of the Helmholtz equation (2.1).

3. Stability analysis using Von Neumann method

Suppose we use Jacobi iteration for solving a system of equations generated by (2.4). The recurrence relation for the Helmholtz equation (2.1) is then given by:

$$\Pi'^{(k+1)}_{i,j} = \frac{1}{(4 - h^2\omega^2)} \left[\Pi'^{(k)}_{i+1,j} + \Pi'^{(k)}_{i-1,j} + \Pi'^{(k)}_{i,j+1} + \Pi'^{(k)}_{i,j-1} \right] \tag{3.1}$$

Let us define the error at the kth iteration as follows:

$$e^{(k)}_{i,j} = \Pi'^{(k)}_{i,j}(Exact) - \Pi'^{(k)}_{i,j}(Approximated) \tag{3.2}$$

Since the exact solution satisfies the relation (3.1), so is the error. Hence,

$$e_{i,j}^{(k+1)} = \frac{1}{(4 - h^2\omega^2)} \left[e_{i+1,j}^{(k)} + e_{i-1,j}^{(k)} + e_{i,j+1}^{(k)} + e_{i,j-1}^{(k)} \right] \tag{3.3}$$

The error term is represented in the form

$$e_{l,m}^{(k)} = A_{pq} \sin \frac{p\pi l}{m} \sin \frac{q\pi m}{m}, \ 1 \leq p, \ q \leq m - 1 \tag{3.4}$$

A_{pq} is an arbitrary constant. Following from (3.4), we have the following expressions:

$$\begin{aligned}
e_{i+1,j}^{(k)} &= A_{pq} \sin \frac{p\pi(i + 1)}{m} \sin \frac{q\pi j}{m} \\
&= A_{pq} \sin \frac{q\pi j}{m} \left[\sin \frac{p\pi i}{m} \cos \frac{p\pi}{m} + \cos \frac{p\pi i}{m} \sin \frac{p\pi}{m} \right]
\end{aligned} \tag{3.5}$$

$$\begin{aligned}
e_{i-1,j}^{(k)} &= A_{pq} \sin \frac{p\pi(i - 1)}{m} \sin \frac{q\pi j}{m} \\
&= A_{pq} \sin \frac{q\pi j}{m} \left[\sin \frac{p\pi i}{m} \cos \frac{p\pi}{m} - \cos \frac{p\pi i}{m} \sin \frac{p\pi}{m} \right]
\end{aligned} \tag{3.6}$$

$$\begin{aligned}
e_{i+1,j}^{(k)} + e_{i-1,j}^{(k)} &= 2A_{pq} \sin \frac{q\pi i}{m} \sin \frac{p\pi i}{m} \cos \frac{p\pi}{m} \\
&= 2 \cos \frac{p\pi}{m} \left[A_{pq} \sin \frac{p\pi i}{m} \sin \frac{q\pi j}{m} \right] \\
&= 2 \cos \frac{p\pi}{m} e_{i,j}^{(k)}
\end{aligned} \tag{3.7}$$

In the selfsame way,

$$\begin{aligned}
e_{i+1,j}^{(k)} + e_{i-1,j}^{(k)} &= 2A_{pq} \cos \frac{q\pi}{m} \sin \frac{p\pi i}{m} \sin \frac{q\pi j}{m} \\
&= 2 \cos \frac{q\pi}{m} e_{i,j}^{(k)}
\end{aligned} \tag{3.8}$$

Substituting equations (3.7) and (3.8) into equation (3.3) yields:

$$\begin{aligned}
e_{i,j}^{(k+1)} &= \frac{1}{(4 - h^2\omega^2)} \left[2 \cos \frac{p\pi}{m} + 2 \cos \frac{q\pi}{m} \right] e_{i,j}^{(k)} \\
&= \xi e_{i,j}^{(k)} \ where \ \xi = \frac{1}{(4 - h^2\omega^2)} \left[2 \cos \frac{p\pi}{m} + 2 \cos \frac{q\pi}{m} \right]
\end{aligned} \tag{3.9}$$

ξ is the propagating factor. For stability;

$$|\frac{e_{i,j}^{(k+1)}}{e_{i,j}^{(k)}}| = |\xi| < 1 \tag{3.10}$$

This implies that

$$|\frac{1}{(4 - h^2\omega^2)} \left[2 \cos \frac{p\pi}{m} + 2 \cos \frac{q\pi}{m} \right]| < 1 \tag{3.11}$$

Let $p = q = 1$. Therefore,

$$-1 < \left(\frac{4}{4 - \omega^2 h^2} \cos \frac{\pi}{m} \right) < 1 \tag{3.12}$$

Equation (3.12) is the condition for stability of the Helmholtz equation (2.1). It shows that the wave number, ω, and the spatial step-size, h, determine the stability of the numerical scheme. To guarantee stability, the wave number and the spatial step-sizes should be chosen such that equation (3.12) is satisfied.

4. A sample problem

To analyse efficiency of various iterative methods, we make use of the following example.

Let us seek a numerical solution to the Helmholtz equation

$$\nabla^2 \Pi'(x, y) + \omega^2 \Pi'(x, y) = \Phi(x, y), \tag{4.1}$$

with boundary condition

$$\Pi'(x, y) = \ln \left[(x + 1)^2 + y^2 \right] \tag{4.2}$$

Let $\Phi(x, y) = \dfrac{1}{4}(x + y)$, $\Delta x = \Delta y = 0.25$ and $\omega = 1$. Furthermore, let this problem be solved on a unit square, $0 \leq x, y \leq 1$. Central difference discretisation of this problem, as illustrated in section 2 above, yields a system of equations in the form

$$\mathbf{A}\Pi' = \mathbf{b} \tag{4.3}$$

That is;

$$
\begin{pmatrix}
3.9375 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
-1 & 3.9375 & -1 & 0 & -1 & 0 & 0 & 0 & 0 \\
0 & -1 & 3.9375 & 0 & 0 & -1 & 0 & 0 & 0 \\
-1 & 0 & 0 & 3.9375 & -1 & 0 & -1 & 0 & 0 \\
0 & -1 & 0 & -1 & 3.9375 & -1 & 0 & -1 & 0 \\
0 & 0 & -1 & 0 & -1 & 3.9375 & 0 & 0 & -1 \\
0 & 0 & 0 & -1 & 0 & 0 & 3.9375 & -1 & 0 \\
0 & 0 & 0 & 0 & -1 & 0 & -1 & 3.9375 & -1 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 3.9375
\end{pmatrix}
\begin{pmatrix}
\Pi'_1 \\ \Pi'_2 \\ \Pi'_3 \\ \Pi'_4 \\ \Pi'_5 \\ \Pi'_6 \\ \Pi'_7 \\ \Pi'_8 \\ \Pi'_9
\end{pmatrix}
=
\begin{pmatrix}
0.6319117244 \\ 0.9984302162 \\ 2.771030124 \\ 0.4106435513 \\ 0.25 \\ 1.759418983 \\ 1.637270447 \\ 1.491154996 \\ 3.294669267
\end{pmatrix}
\tag{4.4}
$$

In the following section, the system of equations (4.4) is solved using various iterative methods. A brief outline of the principle behind each method is given.

5. Iterative schemes

In this section, we outline and briefly describe various iterative schemes for solving the system of equations (4.4). More study on these methods can be obtained in [5] and [1] from which this summary is obtained.

5.1. Jacobi Iterative (JI) Method. It is given by,

$$\Pi'^{(k+1)} = \mathbf{D}^{-1}\left[\mathbf{b} - (\mathbf{L} + \mathbf{U})\Pi'^{(k)}\right], \ k = 0, \ 1, \ 2, \ ... \tag{5.1}$$

where $\mathbf{D}$, $\mathbf{L}$ and $\mathbf{U}$ are the diagonal, lower and upper matrices, respectively, obtained by decomposition of matrix $\mathbf{A}$. k is the number of iteration.

5.2. Gauss-Seidel Iterative (GSI) Method. This is given by,

$$\Pi'^{(k+1)} = (\mathbf{L} + \mathbf{D})^{-1}\left[\mathbf{b} - \mathbf{U}\Pi'^{(k)}\right], \ k = 0, \ 1, \ 2, \ ... \tag{5.2}$$

The unknowns are similar to those in subsection (5.1) above.

5.3. Successive-Over Relaxation (SOR) Method. This is a modification of the Gauss-Seidel method to accelerate its convergence. It is expressed as follows:

$$\Pi'^{(k+1)} = (\mathbf{D} + \alpha\mathbf{L})^{-1}\left[(1-\alpha)\mathbf{D} - \alpha\mathbf{U}\right]\Pi'^{(k)} + \alpha(\mathbf{D} - \alpha\mathbf{L})^{-1}\mathbf{b}, \ k = 0, \ 1, \ 2, \ ... \tag{5.3}$$

α is called the relaxation factor. It can be shown that convergence is obtained within the range $0 < \alpha < 2$. To accelerate convergence of an already convergent Gauss-Seidel Method, over-relaxation method is used, that is, by choosing α in the interval $1 < \alpha < 2$.

5.4. Conjugate Gradient (CG) Method. This scheme is given by;

$$\Pi'^{(k)} = \Pi'^{(k-1)} + t_k\mathbf{v}^{(k)} \tag{5.4}$$

where $\mathbf{v}^{(k)}$ is a search direction expressed in vector form.

$$t_k = \frac{< \mathbf{v}^{(k)}, \ \mathbf{b} - \mathbf{A}\Pi'^{(k-1)} >}{< \mathbf{v}^{(k)}, \ \mathbf{A}\mathbf{v}^{(k)} >} \tag{5.5}$$

5.5. Bi-Conjugate Gradient Stabilized (BiCGSTAB) Method. This method avoids the often irregular convergence patterns of the Conjugate Gradient Squared method. It computes $i \to \mathbf{Q}_i(\mathbf{A})\mathbf{P}_i(\mathbf{A})\mathbf{r}^{(0)}$ where $\mathbf{Q}_i$ is an ith degree polynomial describing a deepest descent.

5.6. Bi-Conjugate Gradient (BICG) Method. Unlike in the Conjugate Gradient method, the residuals are replaced with relations that are similar but based on $\mathbf{A}^T$ instead of $\mathbf{A}$. Two sequences of residuals are updated.

$$\mathbf{r}^{(i)} = \mathbf{r}^{(i-1)} - \alpha_i\mathbf{A}\mathbf{p}^{(i)} \tag{5.6}$$

and

$$\tilde{\mathbf{r}}^{(i)} = \tilde{\mathbf{r}}^{(i-1)} - \alpha_i\mathbf{A}^T\tilde{\mathbf{p}}^{(i)} \tag{5.7}$$

The sequences of search directions are:

$$\mathbf{p}^{(i)} = \mathbf{r}^{(i-1)} + \beta_{i-1}\mathbf{p}^{(i-1)} \tag{5.8}$$

and

$$\tilde{\mathbf{p}}^{(i)} = \tilde{\mathbf{r}}^{(i-1)} + \beta_{i-1}\tilde{\mathbf{p}}^{(i-1)} \tag{5.9}$$

where $\alpha_i = \dfrac{\tilde{\mathbf{r}}^{(i-1)^T}\mathbf{r}^{(i-1)}}{\tilde{\mathbf{p}}^{(i)^T}\mathbf{A}\mathbf{p}^{(i)}}$ and $\beta_i = \dfrac{\tilde{\mathbf{r}}^{(i)^T}\mathbf{r}^{(i)}}{\tilde{\mathbf{r}}^{(i-1)^T}\mathbf{r}^{(i-1)}}$ respectively.

This ensures the bi-orthogonality relations:

$$\tilde{\mathbf{r}}^{(i)^T}\mathbf{r}^{(j)} = \tilde{\mathbf{p}}^{(i)^T}\mathbf{A}\mathbf{p}^{(i)} = 0 \tag{5.10}$$

$$\text{if } i \neq j$$

5.7. **Quasi-Minimal Residual (QMR) Method.** It is meant to overcome the irregularity of the convergence of BICG method. It solves the reduced tri-diagonal system in a least squares sense. It uses look-ahead techniques to avoid breakdowns that make it more robust than BICG.

5.8. **Generalised Minimal Residual (GMRES) Method.** The iterates of this method are given by;

$$\mathbf{x}^{(i)} = \mathbf{x}^{(0)} + y_1\mathbf{v}^{(1)} + ... + y_i\mathbf{v}^{(i)} \tag{5.11}$$

The coefficients y_k are chosen to minimise the residual norm $\|\mathbf{b} - \mathbf{A}\mathbf{x}^{(i)}\|$.

6. RESULTS

The solution of the sample 2-dimensional Helmholtz equation has been summarised in Table 1 and Figure 1 for the different methods used.

TABLE 1. A summary of the number of iterations and CPU-Time for methods used

| Method	JI	GSI	SOR	CG	BiCGSTAB	BICG	QMR	GMRES
No of Iterations	34	17	12	5	2	9	9	3
CPU-Time ×1000	31.2	0	0	0	0	31.2	46.8	15.6

1

---
[1]The zero value in Table 1 implies a negligible value

FIGURE 1. A graph showing the number of iterations and CPU-time of different methods
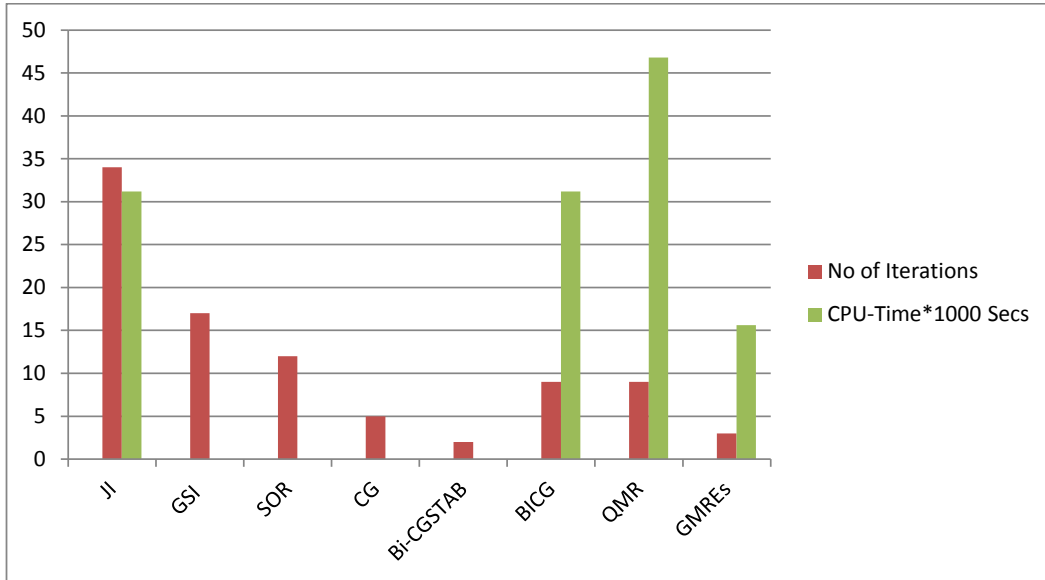
The Helmholtz equation can be visualized using the Graphical User Interface in MATLAB. The following is an output of the Helmholtz equation with $\omega = 1$.

FIGURE 2. Visualization of the Helmholtz equation in MATLAB
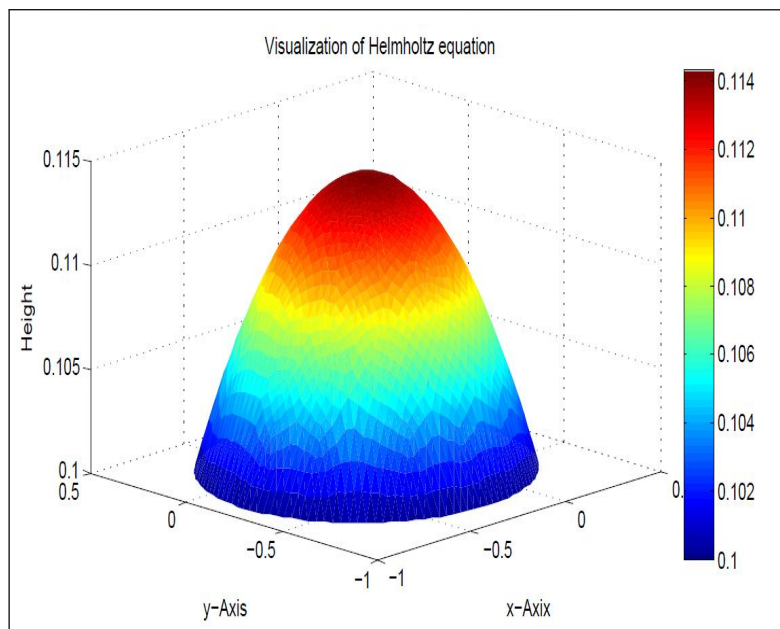
7. Conclusion

Stability analysis of the Helmholtz equation using Von Neumann method shows that stability is guaranteed when the wave number and the spatial step size are chosen such that;

$$-1 < \left(\frac{4}{4 - \omega^2 h^2} \cos \frac{\pi}{m} \right) < 1 \tag{7.1}$$

Solving the Helmholtz equation iteratively, using different methods, shows that the iterations converge faster by Bi-Conjugate Gradient Stabilized Method followed by the Generalised Minimal Residual Method. The least convergent method is the Jacobi Iterative Method. While the Quasi-Minimal Residual Method converges relatively faster than the Jacobi Iterative Method, it takes longer CPU-time than all other methods. It is therefore not computationally efficient. The most computationally efficient method for solving the Helmholtz equation is therefore Bi-Conjugate Gradient Stabilized Method followed by Generalised Minimal Residual Method.

References

[1] Barrett, R., et al, Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, SIAM, Philadelphia PA, 1994.
[2] Buckeridge S., Numerical Solution of Weather and Climate Systems, Ph. D Thesis, University of Bath, 2010.
[3] Diamantakis M., Davies T., and Wood N., An Iterative time-stepping scheme for the Met Office's semi-implicit semi-Lagrangian non-hydrostatic model, Quarterly Journal of The Royal Meteorological Society, Q.J.Meteorol. Soc. (2007)133: 997-1011, Met Office, Exeter, UK.
[4] Norma A., Asnida G., Hafizah F., Elliptic-Helmholtz Equation for early Detection of Breast Cancer Growth using MATLAB Computing, Ibnu Sina Institute, Department of Mathematics Universiti Teknologi Malaysia, 2013.
[5] Rizwan B., Introduction to Numerical Analysis Using Matlab, U.S.A, 2007.

Received: November 1, 2013