Investigation of Triangle Element Analysis for the Solutions of 2D Poisson Equations via AOR method

Mohd Kamalrulzaman Md Akhir, Jumat Sulaiman

Faculty of Science and Natural Resources, Universiti Malaysia Sabah

Locked Bag 2073, 88999 Kota Kinabalu Sabah, Malaysia

ABSTRACT

In earlier studies of iterative approaches, the accelerated over relaxation (AOR) method has been pointed out to be relatively faster than the existing successive over relaxation (SOR) and Gauss Seidel (GS) methods. Due to the effectiveness of this method, the foremost goal of this paper is to demonstrate the use of the AOR method, together with triangle element solutions, based on the Galerkin scheme method. The effectiveness of this method has been shown via the results of numerical experiments, which have been logged and examined. The findings reveal that the AOR method is superior compared to the existing SOR and GS methods.

Keywords

Poisson; AOR method; Galerkin scheme; Triangle Element, iterative methods.

1. INTRODUCTION

Consider the 2D Poisson equation to be given as follows:

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} = f(x,y), \quad (x,y) \in [a,b] \times [a,b] \tag{1}$$

with the dirichlet boundary conditions:

$$U(x,a) = g_1(x), \quad a \le x \le b, \quad U(x,b) = g_2(x), \quad a \le x \le b,$$
$$U(a,y) = g_3(y), \quad a \le x \le b, \quad U(b,y) = g_4(x), \quad a \le y \le b.$$

whereas, $f(x,y)$ is a given function with abundant smoothness. There are numerous studies in the related literature on numerical methods for solving problem (1) [2,11,14]. The applications of numerical methods for problem (1) mostly lead to sparse linear systems, and sometimes the condition number of the corresponding matrices is large. The computational complexity of the matrix and solving the corresponding linear system is huge when the mesh size of the matrix is large. Consequently, in this paper, implementation and performance of the weighted parameter approach based on AOR method with Galerkin scheme for solving FE approximation equations arise from the discretization of problem (1). Generally, the basic concept of the AOR approach is to speed-up the computational time by reducing the number of iterations of the solution method. In order to facilitate formulating the triangle element approximation equations for problem (1), our next discussion focuses on uniform node points only, as shown in Figure 1.
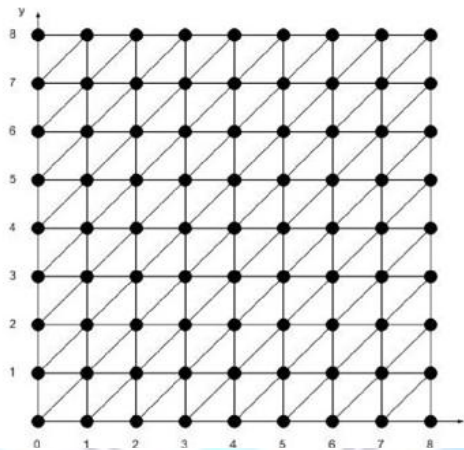

Figure 1: Show the solution domain Ω of triangle elements for the full-sweep cases at *n*=8.

Based on Figure 1, the solution domain needs to be discretized uniformly in both *x* and *y* directions with a mesh size *h*, which is defined as follows:

$$h = \frac{b-a}{n}, \quad m = n+1. \tag{2}$$

where *n* is an arbitrary positive integer. Similarly, using the same concept as that in the full-sweep design, applied to finite difference (FD) methods [3], FE networks consist of several triangle elements, in which each triangle element involves three solid node points of type ● only. Thus, the implementation of the full-sweep idea is carried out onto the node points of the same type until the iterative convergence test is reached [9].

The framework of this paper is organized in the following manner. An implementation of the FE method, which is based on the triangle element being used to discretize problem (1), is presented in Section 2. This is followed by the formulation of the proposed method in Section 3. Numerical results of the tested methods and concluding remarks are summarized in Sections 4 and 5 respectively.

2. TRIANGLE ELEMENT APPROXIMATIONS

As mentioned in the previous section, this paper discusses the use of the AOR method by using a linear FE approximation equation based on the Galerkin scheme to solve 2D Poisson equations. By considering three node points of type ● only,
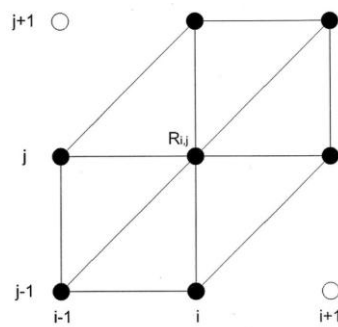
Figure 2: The definition of the hat function $R_{i,j}(x, y)$, of triangle elements at the solution domain.

the general approximation of the function, $U(x, y)$ in the form of an interpolation function for an arbitrary triangle element e is given by [5,6]:

$$\tilde{U}^{[e]}(x, y) = N_1(x, y)U_1 + N_2(x, y)U_2 + N_3(x, y)U_3 \tag{3}$$

and the shape functions $N_k(x, y), \mathrm{k} = 1, 2, 3$ can generally be shown as:

$$\det \mathrm{A} = x_1(y_2 - y_3) + x_2(y_3 - y_1) + x_3(y_1 - y_2), \tag{4}$$

where

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} x_2 y_3 - x_3 y_2 \\ x_3 y_1 - x_1 y_3 \\ x_1 y_2 - x_2 y_1 \end{bmatrix}, \quad \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} y_2 - y_3 \\ y_3 - y_1 \\ y_1 - y_2 \end{bmatrix}, \quad \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} x_3 - x_2 \\ x_1 - x_3 \\ x_2 - x_3 \end{bmatrix},$$

In line to this, the first order partial derivatives of the shape functions towards x and y are given, respectively, as:

$$\left. \begin{aligned} \frac{\partial}{\partial x}\left(N_k(x, y)\right) &= \frac{b_k}{\det A}, \\ \frac{\partial}{\partial y}\left(N_k(x, y)\right) &= \frac{c_k}{\det A}, \end{aligned} \right\}, \quad k = 1, 2, 3 \tag{5}$$

In Figure 2, the definition of the hat function, $R_{r,s}(x, y)$ in the solution domain is easily shown. Based on the distribution of the hat function, $R_{r,s}(x, y)$, the approximation of the functions, $U(x, y)$ and $f(x, y)$, for the entire domain, are defined, respectively, as [10] follows:

$$\tilde{U}(x, y) = \sum_{r=0}^{m}\sum_{s=0}^{m} R_{r,s}(x, y)U_{r,s} \tag{6}$$

$$\tilde{f}(x, y) = \sum_{r=0}^{m}\sum_{s=0}^{m} R_{r,s}(x, y)f_{r,s} \tag{7}$$

Thus, Eqs. (6) and (7) are approximate solutions for problem (1). Therefore, let us consider the Galerkin scheme to be defined as follows:

$$\iint_D R_{i,j}(x, y)E_{i,j}(x, y) = 0, \quad i.j = 0, 1, 2, \dots, \mathrm{m} \tag{8}$$

where $E(x,y) = \dfrac{\partial^2 U}{\partial x^2} + \dfrac{\partial^2 U}{\partial y^2} - f(x,y)$ is a residual function. By applying the Green theorem, Eq. (6) can be shown as follows:

$$\int_\lambda \left(-R_{i,j}(x,y)\frac{\partial u}{\partial y}dx + R_{i,j}(x,y)\frac{\partial u}{\partial x}dy \right)$$
$$-\int_a^b\int_a^b \left(\frac{\partial R_{i,j}(x,y)}{\partial x}\frac{\partial u}{\partial x} + \frac{\partial R_{i,j}(x,y)}{\partial y}\frac{\partial u}{\partial y} \right)dxdy = F_{i,j} \tag{9}$$

where

$$-\sum\sum K_{i,j,r,s}^* U_{r,s} = \sum\sum C_{i,j,r,s}^* f_{r,s} \tag{10}$$

where

$$K_{i,j,r,s}^* = \int_a^b\int_a^b \left(\frac{\partial R_{i,j}}{\partial x}\frac{\partial R_{r,s}}{\partial x} \right)dxdy + \int_a^b\int_a^b \left(\frac{\partial R_{i,j}}{\partial y}\frac{\partial R_{r,s}}{\partial y} \right)dxdy$$

$$C_{i,j,r,s}^* = \int_a^b\int_a^b \left(R_{i,j}(x,y)R_{r,s}(x,y) \right)dxdy$$

The linear system in Eq. (10) can actually be easily expressed in stencil form, respectively, as follows:

$$\begin{bmatrix} & 1 & \\ 1 & -4 & 1 \\ & 1 & \end{bmatrix}U_{i,j} = \frac{h^2}{12}\begin{bmatrix} & 1 & 1 \\ 1 & 6 & 1 \\ 1 & 1 & \end{bmatrix}f_{i,j} \tag{11}$$

To expedite the implementation of AOR method, illustration of the stencil forms in Eq. (11) can be visualized through the computational molecule, as shown in Figure 3. Indeed the computational molecule comprises seven solid node points in formulating their approximation equations. Subsequently, two of its coefficients have a value of zero. This computational molecule for the Galerkin triangle FE scheme is in a form that is similar to the existing five point FD method [1]. By taking account of this computational molecule, further explanation of the AOR method is presented on the manner used to construct its formulation in the next section.
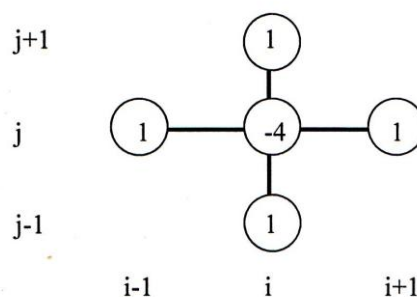

Figure 3: Computational molecule of the Galerkin Triangle FE scheme.

3. THE AOR METHOD

The AOR method was introduced in [7]. Generally, let us consider the numerical solution of the linear system to be as follows:

$$A\underline{U} = \underline{f}, \tag{12}$$

where $A \in \square^{n,n}$ represents nonsingular, sparse matrices with non-vanishing diagonal entries, and $\underline{U}, \underline{f} \in \square^{n,n}$ with $\underline{U}$ to be determined. Let us study the AOR method. Let $A \in \square^{n,n}$ be a one-cycle and consistent orderings ordered matrix of the form:

$$A = \begin{bmatrix} D & U \\ U^T & D \end{bmatrix} \tag{13}$$

where $U \in \square^{n1,n2}, U^T \in \square^{n2,n1}$, and $D \in \square^{n1,n1}$ are diagonal nonsingular matrices, respectively, with $n_1 + n_2 = n.$ Again, let A be defined as follows:

$$A = D - L - V \tag{14}$$

where

$$D = \begin{bmatrix} D & 0 \\ 0 & D \end{bmatrix}, \quad L = \begin{bmatrix} 0 & 0 \\ -U^T & 0 \end{bmatrix}, \quad V = \begin{bmatrix} 0 & -U \\ 0 & 0 \end{bmatrix}$$

The Jacobi matrix is

$$B = D^{-1}(L + V) = \underline{L} + \underline{V} \tag{15}$$

with

$$\underline{L} = D^{-1}L = \begin{bmatrix} 0 & 0 \\ \underline{L} & 0 \end{bmatrix}, \quad \underline{V} = D^{-1}V = \begin{bmatrix} 0 & \underline{V} \\ 0 & 0 \end{bmatrix}.$$

where $\underline{L} = -D^{-1}U^T$ and $\underline{V} = -D^{-1}U$. In general, the AOR iterative scheme can be defined as follows:

$$L_{r,w} = rL(I - rD^{-1}L)^{-1}\left[(1-w)I + (w-r)L + wD^{-1}V\right] \tag{16}$$

Theoretically, applying Eq. (12) to each point (x_i, y_j) leads to a linear system with coefficient matrix A given by:

$$A = \begin{bmatrix} D_0 & V_0 & & & \\ L_0 & D_0 & V_0 & & \\ & L_0 & D_0 & \ddots & \\ & & \ddots & \ddots & V_0 \\ & & & L_0 & D_0 \end{bmatrix}_{(n-1)^2 \times (n-1)^2} \tag{17}$$

where

$$D_0 = \begin{bmatrix} R_0 & R_1 & & & \\ R_1^T & R_0 & R_1 & & \\ & R_1^T & R_0 & \ddots & \\ & & \ddots & \ddots & R_1 \\ & & & R_1^T & R_0 \end{bmatrix}_{(n-1)\times(n-1)}, \quad V_0 = \begin{bmatrix} R_2 & & & & \\ & R_2 & & & \\ & & R_2 & & \\ & & & \ddots & \\ & & & & R_2 \end{bmatrix}_{(n-1)\times(n-1)},$$

$$L_0 = \begin{bmatrix} R_2^T & & & & \\ & R_2^T & & & \\ & & R_2^T & & \\ & & & \ddots & \\ & & & & R_2^T \end{bmatrix}_{(n-1)\times(n-1)}.$$

and the submatrices R_0, R_1 and R_2 are given by:

$$R_0 = \begin{bmatrix} 4 & -1 & & -1 \\ -1 & 4 & -1 & \\ & -1 & 4 & -1 \\ -1 & & -1 & 4 \end{bmatrix}, \quad R_1 = \begin{bmatrix} 0 & -1 & & \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ & & & 0 \end{bmatrix}, \quad R_2 = \begin{bmatrix} & & & 0 \\ & & 0 & \\ & -1 & & \\ -1 & & & \end{bmatrix}.$$

respectively. Since the coefficient matrices in Eq. (17) represent a pointwise pentadiagonal with non-vanishing diagonal element, it has the property A and $\pi -$ Consistently Ordered, as prescribed in [4,12,13]. Hence, the theory of AOR method is also valid in triangle element approaches. The AOR method generally implicates dual parameters, r and w, where all the communal existing methods such as Jacobi, GS and SOR become common cases of this method when the parameters are given certain values. For example, when $w = 1$ and $r = 0$, we attain the Jacobi method. If $w = r = 1$, we obtain the GS method. If $w = r$, we obtain the SOR method.

In order to perform the AOR method, the solution domain is divided as shown in Figure. 1. Conferring to Eq. (16), the AOR method over the linear system Eq. (17) can be stated as follows:

$$U_{i,j}^{(k+1)} = \frac{r}{4}\left(U_{i-1,j}^{(k+1)} - U_{i-1,j}^{(k)} + U_{i,j-1}^{(k+1)} + U_{i,j-1}^{(k)}\right)$$

$$+ \frac{w}{4}\left(U_{i+1,j+1}^{(k)} + U_{i-1,j-1}^{(k)} + U_{i+1,j-1}^{(k)} + U_{i-1,j+1}^{(k)} - F_{i,j}\right) + (1-w)U_{i,j}^{(k)} \tag{18}$$

where

$$F_{i,j} = \frac{h^2}{12}\left(f_{i-1,j} + f_{i+1,j} + f_{i,j-1} + f_{i,j+1} + f_{i-1,j-1} + f_{i+1,j+1} + 6f_{i,j}\right)$$

Eq. (18) allows us to iterate through half of the points lying on the h-grid. Consequently, the iteration can be carried out autonomously, involving only this type of point, as prescribed in the following algorithm:

Discretize the solution domain into point of type ●, as shown in Figure. 1.

1. Perform iterations (using Eq. (18)).

2. Within the interval of ±0.1 from the value found in Step 2, define the optimal ω opt with a precision of 0.01 by choosing consecutive values for which k is minimal; r is taken to be equal to ω.

3. Perform experiments using the value of ω opt, and choose consecutive values of r with a precision of 0.01 within the interval of ±0.1 from the ω opt.

4. Define the value r opt for which k is minimal.

5. Display the approximate solutions.

4. NUMERICAL RESULTS

In this section, the proposed algorithm was tested on the following model, following the 2D Poisson equation:

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} = -\left(\cos(x+y) + \cos(x-y)\right) \tag{19}$$

$$U(x,0) = \cos x, \quad U\left(x, \frac{\pi}{2}\right) = 0,$$

$$U(0, y) = \cos y, \quad U(\pi, y) = -\cos y.$$

The exact solution is given by:

$$U(x, y) = \cos(x)\cos(y)$$

Through the experiments, three parameters were observed, namely, the number of iterations (k), the maximum absolute error (Abs. Error) and the execution time, t (in seconds). Three methods (GS, SOR and AOR) were tested on several mesh sizes (i.e., 284, 308, 332 and 356). In the course of implementation of the proposed methods, the value of the tolerance error was considered $\varepsilon = 10^{-10}$. The C++ programming lanauge was used for constructing the program, and the program was execiuted on a PC with an Intel(R) Core (TM) i7 CPU 860@3.00 Ghz, and 6.00GB RAM. The operation system used was Windows 7, along with a installation Borland C++ compiler version 5.5. Based on the given example, all of the results of the numerical experiments were recorded through the implementation of the proposed methods in Table 1, while Table 2 describes the depreciation percentage of the number of iterations and execution time for the AOR and SOR methods compared to the GS method.

Table 1: Comparison of a number of iterations, execution time (seconds) and maximum absolute error for the iterative methods.

N	Methods	r	w	k	t	Abs. Error
	GS	-	-	115954	267.61	1.0913e-6
284	SOR	-	1.952	3428	11.97	3.0217e-7
	AOR	1.989	1.979	1890	9.43	2.8268e-7
	GS	-	-	134823	366.50	1.1936e-6
308	SOR	-	1.949	4288	17.69	2.6481e-7
	AOR	1.989	1.987	1939	11.75	2.4128e-7
	GS	-	-	154979	490.68	1.3165e-6
332	SOR	-	1.939	5960	28.58	2.4124e-7
	AOR	1.986	1.991	2668	18.84	2.0723e-7
	GS	-	-	176045	638.22	1.4576e-6
356	SOR	-	1.939	6815	37.63	2.1950e-7
	AOR	1.987	1.993	3032	24.70	1.8023e-7

Table 2: Reduction percentages of the number of iterations and execution time for the iterative methods compared with GS method.

Methods	k	t
SOR	96.13–97.04	94:10–95:49
AOR	96.82–98.36	96.12–96.44

5. CONCLUSION

The results from the previous section demonstrated that triangle element approximation equations based on Galerkin scheme can easily be presented in Eq. (11). According to the numerical results obtained for the proposed problem, as shown in Tables 1 and 2, they clearly show that applying the AOR method can reduce the number of iterations and execution time compared to the SOR and GS methods. Moreover, approximate solutions for the AOR method are also in good agreement compared to the SOR and GS methods. For future work, this study will be extended to investigate the applications of the half-sweep concept [1] with the AOR method by using FE approximation equations.

REFERENCES

[1] Abdullah, A. R. 1991. The four point explicit decoupled group (EDG) method: A fast Poisson solver, Int. J. Comp. Math. **38**: 61–70.

[2] Belytschko, T., Krongauz, Y., Organ, D., Fleming, M. and Krysl, P. 1996. Meshless methods:An overview and recent developments, Computer Methods in Applied Mechanics AndEngineering, **139**: 3–47.

[3] Evans, D. J. 1997. Group Explicit Methods for the Numerical Solutions of Partial Differential Equations, Gordon and Breach Science Publishers, Australia.

[4] Evans, D. J. and Martins, M. M. 1994. The AOR method for AX-XB = C, Int. J. Comp. Math. **52**: 75–82.

[5] Fletcher, C. A. J. 1978. The Galerkin method: An introduction, in J. Noye (ed.), Numerical Simulation of Fluid Motion, North-Holland Publishing Company, Amsterdam, pp. 113–170.

[6] Fletcher, C. A. J. 1984. Computational Galerkin method, Springer Series in Computational Physics, Springer-Verlag, New York.

[7] Hadjidimos, A. 1978. Accelerated over relaxation method, Mathematics of Computation **32**: 149–157.

[8] Lewis, P. E. and Ward, J. P. 1991. The Finite Element Method: Principles and Applications, Addison-Wesley Publishing Company, Wokingham.

[9] Saad, Y. 1996. Iterative method for sparse linear systems, Thomas, Boston.

[10] Vichnevetsky, R. 1981. Computer Methods for Partial Differential Equations, Vol I, Prentice-Hall, New Jersey.

[11] Yagawa, G. and Furukawa, T. 2000. Recent developments of free mesh method, Int. J. for Num. Meth. in Eng. **47**: 1419–1443.

[12] Young, D. M. 1954. Iterative methods for solving partial difference equations of elliptic type, Trans. Amer. Math. Soc. **76**: 92–111.

[13] Young, D. M. 1971. Iterative solution of large linear systems, Academic Press, London.

[14] Zhu, T. 1999. A New Mesh less Regular Local Boundary Integral Equation (MRLBIE) approach. Int. J. for Num. Meth. in Eng., 46 (1999) 1237-125.