ISSN 2462-2052 Journal of Science and Mathematics Letters
DOI: https://doi.org/10.37134/jsml.vol11.1.1.2023 Volume 11, Issue 1, 1-9, 2023

RESEARCH ARTICLE

Agent Navigation based on Boundary Value Problem Using Iterative Methods

Farhah Athirah Musli and Azali Saudi*

Faculty of Computing and Informatics, Universiti Malaysia Sabah,
88400 Kota Kinabalu, Malaysia
*Corresponding author: azali@ums.edu.my

Received: 15 June 2022; **Accepted:** 25 September 2022; **Published:** 28 February 2023

ABSTRACT

This paper presents the simulation of numerical solutions to the navigational problem of an agent traveling safely in its environment. The approach is based on the numeric solutions of the boundary value problem (BVP) that generate harmonic potential fields through a differential equation whose gradient represents navigation routes to the destination. Two methods, namely KSOR and KAOR, were tested to solve the BVP. KSOR and KAOR are variants of the standard SOR and AOR methods, respectively. In this work, the KSOR and KAOR methods were used to solve the BVP by applying Laplace's equation to obtain harmonic functions. The generated harmonic functions are then utilized by the searching algorithm to find a smooth navigational route for an agent to travel in its environment without colliding with any obstacles. The numerical results from the solutions of BVP demonstrate that the KAOR provides a faster execution time with fewer iterations compared to the KSOR method.

Keywords: KSOR, KAOR, Harmonic function, navigation, boundary value problem, Laplace's equation

1. INTRODUCTION

Navigation problem is known as one of the most challenging topics in robotics and automated applications. The main aim of solving the navigation problem is basically on finding the best route for an object to move without colliding with any obstacles in the specified environment. There are several concerns on navigation problems such as the efficiency and safety. Firstly, the efficiency of the method used is considered the most significant factor as it will act as a guide to find the destination in a short time. Therefore, the generated route should not cause the agent from stuck at the local minima and tax too much time for unnecessary moves. Next, the safeness of the route is one of the critical issues in route navigation. The generated route should be a collision-free route where it can avoid all known obstacles inside the environment.

Potential field is a commonly used method to find a safe navigational route for an agent to move safely in its environment (Montiel, et. al., 2015). Potential field has the ability to adapt toward unknown scenarios by understanding the current state of the environment (Sabudin et al., 2016). This method, however, suffers from a local minima problem where it cause the agent stuck or not move at all. Hence, in order to overcome this problem, this study employs the harmonic potentials to solve the route navigational problem. Harmonic functions are known as the solution for Laplace's equation. They are known to have several properties that are beneficial in robotics and automated applications. The first implementation of the solution of Laplace's equation into route navigational problem was independently conducted by Connolly et al. (1990) and Akishita et al. (1990).

Various methods have been investigated for computing the harmonic functions, but the most common methods are approaches based on numerical methods. This is mainly due to the capability of fast computing resources and efficient numerical methods in solving the problem. Prior, there were three standard iterative methods used to solve route navigation problems such as Jacobi, Gauss-Seidel (GS), and Successive Over Relaxation (SOR) (Dahalan et al., 2017). In the literature, it was shown that the performance of SOR is noticeably faster than the classical Jacobi and standard GS. Alternatively, Daily and Bevly (2008) used an analytical solution to solve the linear system. The use of the finite element method for solving Laplace's equation was reported by Al-Taweel et al. (2021). A deep learning approach to the problem was presented by Nguyen et al. (2020). The application of BVP for path planning was demonstrated by Prestesy and Idiartz (2010). In the work by Wray et al. (2016) and Chou et al. (2017), path planning based on log-space harmonic potential was applied. More recently, a harmonic-based solution was also applied to deal with the autonomous robot exploration problem (Grontas et al., 2020). In 2012, Youssef introduced a new variant of SOR called Kaudd Successive Over Relaxation (KSOR). Later, a new version of KSOR was described by Constantinescu et al. (2019). A study on KSOR such as solving integral equation problems was reported by Radzuan et al. (2017).

A variant of KSOR was applied to solve parabolic equations Muhiddin et al. (2020), while faster Half-Sweep KSOR was employed in the previous works to reduce the amount of computation (Suardi et al., 2017; Musli and Saudi, 2019). The Kaudd Accelerated Over Relaxation (KAOR) was later introduced by Youssef and Farid (2015) for solving linear systems. They described the advantages of the KAOR in the choice of optimum parameters. However, the KAOR method has not yet been tested for solving BVP in agent navigation problems. Thus, the main purpose of this study is to examine the feasibility of using the KAOR method to obtain the harmonic functions that would be utilized by the route finding algorithm to produce a smooth path for an agent to move safely in its environment.

2. MATERIALS AND METHODS

2.1. Problem formulation

Harmonic functions are known as the solution of Laplace's equation that is beneficial in solving route navigational problems. One of its major advantages is it is free from local minima problems (Akishita et al., 1990). A harmonic function on a domain is a function that satisfies Laplace's equation

$$\nabla^2 U = \sum_{i=1}^{n} \frac{\partial^2 U}{\partial x_i^2} = 0 \tag{1}$$

where x_i and n correspondingly denote the i-th Cartesian coordinate and the dimension of the environment. In the case of the construction of a virtual environment, the boundary consists of both the inner and outer walls and all obstacles. By applying Laplace's equation as a constraint, the creation of a local minimum inside the virtual environment is avoided, since the harmonic functions fulfill the min-max principle. Thus, the only critical points allowed to occur are saddle points where the gradients from all directions are zero.

Navigational route problems for an agent in a virtual environment can be described as a steady-state heat transfer problem. Outer and inner boundary walls and all obstacles inside the virtual environment are treated as heat sources whilst the goal point is fixed as a heat sink that pulls the heat in. This heat-transfer process creates a temperature distribution where the heat moves from higher heat sources to the heat sink with lower temperatures. By using the above analogy, the solutions of Laplace's equation are applied to constrain the harmonic potential distribution in the virtual environment. Through an iterative process, harmonic potentials are computed until the specified convergence criterion is satisfied. The general idea is to trace the distribution of harmonic functions where the potentials flow from higher to lower values. Then, the gradient descends search strategy can be applied to find the navigational route from an arbitrary start point to the lowest potential value at the goal point.

In the simulation, a point inside the virtual environment represents the virtual agent. Meanwhile, the virtual environment itself is represented in two-dimensional rectangular outer boundary walls containing various shapes of inner walls and obstacles. The type of virtual environment selected in this simulation is static maps taken from the study reported in (de Silva et al., 2002). The simulations considered two maps of several sizes.

2.2. The KSOR method

The KSOR is a variant of the SOR method for solving linear system problems that have been introduced by Youssef (2012). In this study, KSOR is used to obtain the solution to Laplace's equation. Let us consider the two-dimensional Laplace's equation (1) is written as:

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} = 0 \tag{2}$$

The second order central difference scheme is then applied to obtain the five-point approximation that can be written as follows

$$U_{i-1,j} + U_{i+1,j} + U_{i,j-1} + U_{i,j+1} - 4U_{i,j} = 0 \tag{3}$$

Equation (3) is used to implement the GS method which is known as a standard iterative method to solve any linear systems (Mohammed and Rivaie, 2017), Figure 1 illustrates the stencil point of the above difference scheme. Consequently, by adding a weighted parameter w, the GS iterative scheme can be deduced into SOR iterative method. Therefore, the iterative scheme of the SOR method is obtained and can be written as:

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

where the value of w is defined as $w \in (2, 0)$. Note that if $w = 1$, the SOR method returns to the standard GS method (Hadjidimos, 2000).

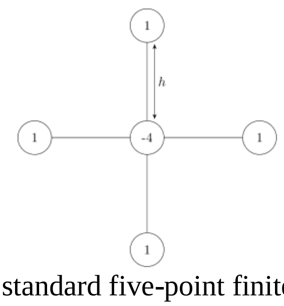

Figure 1. The stencil point of the standard five-point finite difference approximation scheme

ISSN 2462-2052
DOI: https://doi.org/10.37134/jsml.vol11.1.1.2023

The relaxation parameter plays an important role in both SOR and KSOR methods. However, the relaxation parameter of KSOR is defined $w^* \in R-[-2,0]$. Considering a linear system of the form $AX = b$. Matrix A can be decomposed into $A = D - L - U$, where D is a diagonal part of matrix A, and matrices L and U are the strictly lower and upper triangular parts of matrix A, respectively. Accordingly, the KSOR method can be written in matrix form as:

$$X^{[n+1]} = T_{KSOR}X^{[n]} + [(1+w^*)D - w^*]^{-1}w^*b$$
$$T_{KSOR} = [(1+w^*)D - w^*L]^{-1}(D + w^*U).$$

The convergence of the KSOR method is given below:

Theorem 1: Let $A \in R^{m\times m}$ with $a_{ii\neq 0}$. Then $\rho(T_{KSOR}) \geq \frac{1}{|1+w^*|}$, which implies that the KSOR method can converge for all $w^* \in R-[-2,0]$.
Proof: for all, $w^* \neq -1$, we obtain:

$$\begin{aligned}
\det(T_{KSOR}) &= \det((1+w^*)D - w^*L)^{-1}(D+w^*U) \\
&= \det(((1+w^*)D - w^*L)^{-1}\det(D+w^*U)) \\
&= \frac{1}{\det((1+w^*)D - w^*L)}\det(D+w^*U) \\
&= \frac{1}{\det((1+w^*)D)}\det(D+w^*U) \\
&= \frac{1}{(1+w^*)^m\det(D)}\det(D+w^*U) \\
&= \frac{1}{(1+w^*)^m}\det(D)^{-1}\det(D+w^*U) \\
&= \frac{1}{(1+w^*)^m}\det(I+w^*D^{-1}U) \\
&= \frac{1}{(1+w^*)^m}
\end{aligned}$$

Since $\det(T_{KSOR}) = \prod_{j=1}^{m}\beta_j$ where β_j is the eigenvalues of the iteration matrix, T_{KSOR}, accordingly we obtain:

$$\left|\prod^{m}\beta_j\right| = |\det(T_{KSOR})| = \left|\frac{1}{(1+w^*)^m}\right| \leq \max\left|\beta_j\right|^m.$$

Thus $(T_{KSOR}) \geq \frac{1}{|1+w^*|}$. For convergence, we must have $1 > \rho(T_{KSOR}) \geq \frac{1}{|1+w^*|}$ and this gives $w^* \in R-[-2,0]$.

Theorem 2: The KSOR method is completely consistent with the system (1) for all values of the relaxation parameter $w^* \in R-[-1,0]$.

Proof: The proof is given in the definition described in (Young, 1971).

Further details on the convergence proof of KSOR are described in (Youssef and Taha, 2013). The relaxation parameter for the KSOR method is less sensitive compared to the SOR. This relaxation parameter is used to control the spectral radius of the iteration matrices which would significantly affect the convergence rate. The iterative scheme for the KSOR is given as:

$$U_{i,j}^{(k+1)} = \frac{1}{1+w^*} U_{i,j}^{(k)} + \frac{w^*}{4(1+w^*)} \left(U_{i-1,j}^{(k+1)} + U_{i+1,j}^{(k)} + U_{i,j-1}^{(k+1)} + U_{i,j+1}^{(k)} \right). \tag{5}$$

2.3. *The KAOR method*

Apart from the SOR method, Hadjidimos (1978) developed a new variant called AOR iterative scheme that can be written as:

$$U_{i,j}^{(k+1)} = \frac{w}{4} \left[U_{i-1,j}^{(k)} + U_{i+1,j}^{(k)} + U_{i,j-1}^{(k)} + U_{i,j+1}^{(k)} \right] + \frac{r}{4} \left[U_{i-1,j}^{(k+1)} - U_{i-1,j}^{(k)} + U_{i,j-1}^{(k+1)} - U_{i,j-1}^{(k)} \right]. \tag{6}$$

The KAOR method is essentially an extension of the AOR and KSOR methods as described in details by Youssef and Farid (2015). Its iterative scheme can be obtained by adding an accelerated parameter r and is written as:

$$U_{i,j}^{(k+1)} = \frac{1}{1+w^*} U_{i,j}^{(k)} + \frac{w^*}{4(1+w^*)} \left(U_{i-1,j}^{(k)} + U_{i+1,j}^{(k)} + U_{i,j-1}^{(k)} + U_{i,j+1}^{(k)} \right) + \frac{r}{4(1+r)} \left(U_{i-1,j}^{(k+1)} - U_{i+1,j}^{(k)} + U_{i,j-1}^{(k+1)} - U_{i,j+1}^{(k)} \right). \tag{7}$$

From equation (7), it can be seen that the additional parameter extends the range for choosing the optimum value thus reducing the sensitiveness of the spectral radius of the iteration matrix with changes in r and w^* as discussed in (Youssef and Farid, 2015).

2.4. *Path planning algorithm*

Both KSOR and KAOR methods employ traditional full-sweep iteration using a five-point discretization scheme. During the iteration process, all inner nodes in the grid will be computed. Before the iteration procedure, matrix *U* (current harmonic potentials) and matrix *V* (updated harmonic potentials) of the map need to be initialized. Nodes in both matrices that are occupied by obstacles are assigned with high potentials, whereas non-occupied nodes are given with random potential. Target nodes are assigned with the lowest potentials. In each iteration, the harmonic potentials in both matrices are updated using equations (5) and (7) for KSOR and KAOR methods, respectively. The algorithm for KSOR (5) and KAOR (7) iterative methods is described below:

1. Start time, T1
2. w^* = -2.18, r = -2.12 epsilon = 1e-15
3. k = 0
4. **repeat**
5. **for** all nodes **do**
6. **if** node U(i,j) is not occupied **then**
7. Update V(u,j) using equation (5) or (7)
8. **end if**
9. **end for**
10. Update Error
11. k = k + 1
12. **until** Error < epsilon
13. Stop time, T2
14. cputime = T2 - T1
15. **return** V, k, cputime

During the iteration process, only non-occupied nodes are involved in the calculation. All occupied nodes are ignored. Once the convergence criterion is satisfied, the iteration process is stopped. The path searching process is then used to trace a smooth path from any start position to the specified goal position by descending the slope of the harmonic potentials obtained from the iterative methods as described in the above algorithm. From any start position, the path tracing algorithm picks the next node with the lowest potential from the current neighbouring nodes. This process continues until the goal position is reached which is the node with the lowest potential value. The path searching algorithm is described below:

1. Initialize path list, *P*
2. Move to the start point, *Q*
3. **while** node(*S*) is not the target point **do**
4. $Q \longleftarrow \min(S_W, S_E, S_S, S_N)$
5. Push into path list, $P \longleftarrow Q$
6. Move to the next point, *Q*
7. **end while**
8. **return** *P*

where *W*, *E*, *N*, and *S* denote the position of the current four neighbours $(i\text{-}1, j)$, $(i+1, j)$, $(i, j\text{-}1)$, and $(i, j+1)$ of point *S*.

3. RESULTS AND DISCUSSION

The implemented algorithms were evaluated with two indoor maps, to illustrate the impact of the existing KSOR (5) on four different sizes in comparison with the newly proposed KAOR method (7). The numerical simulations were carried out using three grid sizes of 165×150, 330×300, 660×600, and 990×900. Both maps consist of a start point, a goal point, several obstacles, outer rectangular boundary walls, and various setting of inner walls. The solid points in green and red colour denote the start and goal positions, respectively.

The simulations were conducted on a machine with an Intel Core i5-3570K processor running at 3.40GHz clock speed with 16GB memory. For comparison, two factors were recorded such as the number of iterations and computational time for different grid sizes. In addition to that, the convergence criterion is set to a very small value e=1.0^{-15}. Such high precision is used so that the occurrence of a flat area in the final solution is minimized since such a flat area could trigger the route finding process to stop before reaching the goal position. In the KSOR method, the parameter w^* was set to -2.18, while in the KAOR method, the parameters w^* and r were set to -2.18 and -2.12, respectively. Both values were chosen since they gave an optimal convergence rate in the preliminary results.

Figures 2 and 3 illustrate several generated routes for Map 1 and Map 2, respectively. For each map, the path search algorithm successfully generated the route from the starting point (red dot) to the destination point (green dot). The destination point needs to be specified before the harmonic potentials are computed. The initial point can be specified after the harmonic potentials are obtained. As shown in both figures, the generated routes were smooth and safe, since they tended to move away from the obstacles.

Based on Table 1, it is shown that the KAOR iterative method is superior to the KSOR method since it requires less number of iterations. The proposed KAOR method is able to obtain the solutions of Laplace's equation with fewer iterations and faster computational execution than the KSOR on average by approximately 30%. The improvement of the proposed KAOR method is due to faster convergence in computing the harmonic functions.

The computational time of both methods are illustrated in Figure 4. Clearly, the proposed KAOR is more efficient than the KSOR iterative method and thus improves the overall performance of the route finding algorithm. It can also be observed that the execution time increases rapidly as the size of the environment grow.

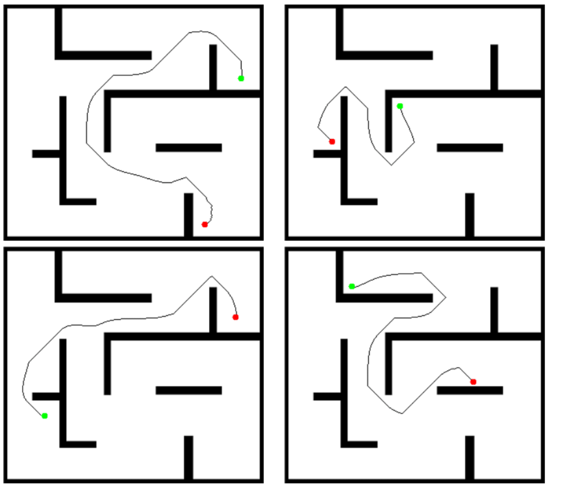

Figure 2. The generated route from several different start and target points on Map 1 of 300×300

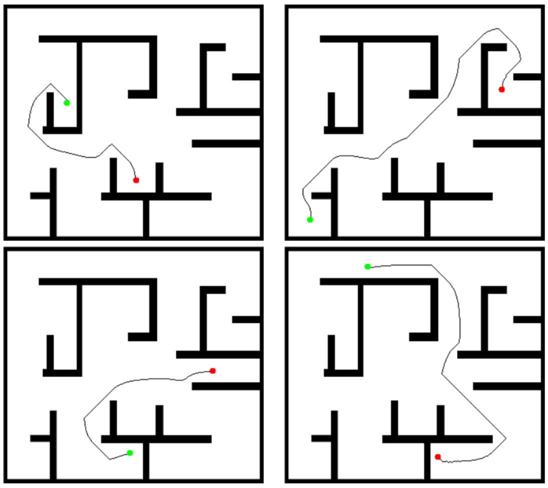

Figure 3. The generated route from several different start and target points on Map 2 of 300×300

Table 1. Iterations and computational time for maps 1 and 2

Map Size	Method	Map 1		Map 2	
		Iterations	Time (seconds)	Iterations	Time (seconds
165 x 150	KSOR	801	0.299	612	0.157
	KAOR	501	0.147	340	0.089
330 x 300	KSOR	2865	2.846	2112	2.063
	KAOR	1897	1.920	1432	1.405
660 x 600	KSOR	11511	46.758	8906	34.974
	KAOR	8183	33.140	6313	25.006
990 x 900	KSOR	24856	234.125	17763	163.793
	KAOR	17676	165.702	12173	112.731

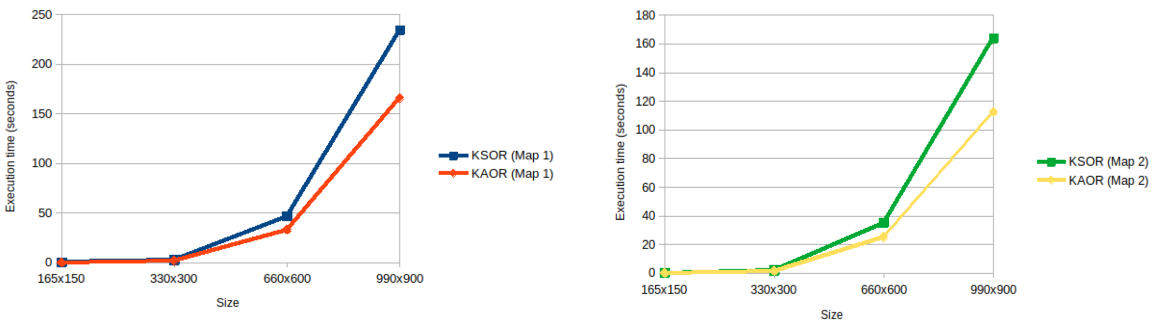

Figure 4. The execution time of the KSOR and KAOR methods on Map 1 and Map 2

4. CONCLUSION

In this paper, the route navigational problem has been solved iteratively via KSOR and KAOR iterative methods. From the observation of numerical results, it is clearly shown that the KAOR iterative method requires fewer iteration numbers compared to the KSOR iterative method. Consequently, it significantly reduced the computational time required to obtain the harmonic potentials. The simulation successfully proved that the generated routes were free from the collision. As an extension to the full-sweep approach, investigation of half-sweep and quarter-sweep iterations will be considered to further speed up the convergence rate of the iteration process, thus improving the performance of the route finding algorithm.

Declaration of Interest
The authors declare that there is no conflict of interest.

Acknowledgement
The authors acknowledge the Universiti Malaysia Sabah for the financial support.

REFERENCES

Akishita S, Kawamura S, Hayashi KI. (1990). Laplace potential for moving obstacle avoidance and approach of a mobile robot. In Japan-USA Symposium on Flexible Automation, A Pacific RIM Conference, p. 139-142.

Al-Taweel A, Dong Y, Hussain S, Wang X. (2021). A weak galerkin harmonic finite element method for laplace equation. *Communications on Applied Mathematics and Computation*, 3(3), 527-543.

Chou HJ, Kuo PL, Liu JS. (2017). Numerical streamline path planning based on log-space harmonic potential function: a simulation study. In Proc. of the 2017 IEEE International Conference on Real-time Computing and Robotics (RCAR), p. 535-542.

Connolly CI, Burns JB, Weiss R. (1990). Path planning using Laplace's equation. In Proc. of the IEEE International Conference on Robotics and Automation, p. 2102-2106.

Constantinescu R, Poenaru RC, Pop F, Popescu PG. (2019). A new version of KSOR method with lower number of iterations and lower spectral radius. *Soft Computing*, 23(22), 11729-11736.

Dahalan AA, Saudi A, Sulaiman J, Din WRW. (2017). Numerical evaluation of mobile robot navigation in static indoor environment via EGAOR Iteration. In Journal of Physics: Conference Series, IOP Publishing, p. 012064.

Daily R, Bevly DM. (2008, June). Harmonic potential field path planning for high speed vehicles. In Proc. of the 2008 American Control Conference, p. 4609-4614.

de Silva Jr EP, Engel PM, Trevisan M, Idiart MA. (2002). Exploration method using harmonic functions. *Robotics and Autonomous Systems*, 40(1), 25-42.

Grontas PD, Vlantis P, Bechlioulis CP, Kyriakopoulos KJ. (2020). Computationally Efficient Harmonic-Based Reactive Exploration. *IEEE Robotics and Automation Letters*, 5(2), 2280-2285.

Hadjidimos A. (1978). Accelerated Overelaxation Method. *Mathematics of Computation*, 32, 149-157, 1978.

Hadjidimos A. (2000). Successive overrelaxation (SOR) and related methods. *Journal of Computational and Applied Mathematics*, 123(1-2), 177-199.

Mohammed F, Rivaie M. (2017). Jacobi-Davidson, Gauss–Seidel and successive over-relaxation for solving systems of linear equations. *Applied Mathematics and Computational Sciences*, 6, 41-52.

Montiel O, Orozco-Rosas U, Sepúlveda R. (2015). Path planning for mobile robots using Bacterial Potential Field for avoiding static and dynamic obstacles. *Expert Systems with Applications*, 42(12), 5177-5191.

Muhiddin FA, Sulaiman, J., and Sunarto, A. (2020). Grünwald implicit solution of one-dimensional time-fractional parabolic equations using HSKSOR iteration. In Journal of Physics: Conference Series, IOP Publishing, p. 2025.

Musli FA, Saudi A. (2019). Agent Navigation via Harmonic Potentials with Half-Sweep Kaudd Successive Over Relaxation (HSKSOR) Method. In Proc. of the 2019 IEEE 9th Symposium on Computer Applications & Industrial Electronics (ISCAIE), p. 335-339.

Nguyen T, Pham B, Nguyen TT, Nguyen BT. (2020). A deep learning approach for solving Poisson's equations. In Proceeding of 12th International Conference on Knowledge and Systems Engineering, p. 213-218.

Prestesy E, Idiartz M. (2010). Computing navigational routes in inhomogeneous environments using bvp path planner. In Proceeding of IEEE/RSJ International Conference on Intelligent Robots and Systems, p. 1427-1432.

Radzuan NZFM, Suardi MN, Sulaiman J. (2017). KSOR iterative method with quadrature scheme for solving system of Fredholm integral equations of second kind. *Journal of Fundamental and Applied Sciences*, 9, 609-623.

Sabudin EN, Omar R, Ckan H, Melor CK. (2016). Potential field methods and their inherent approaches for path planning. *ARPN Journal of Engineering and Applied Sciences*, 11(18), 10801-10805.

Suardi MN, Radzuan NZFM, Sulaiman J. (2017). Cubic B-spline solution of two-point boundary value problem using HSKSOR iteration. *Global Journal of Pure and Applied Mathematics,* 13(11), 7921-7934.

Wray KH, Ruiken D, Grupen RA, Zilberstein S. (2016). Log-space harmonic function path planning. In Proc. of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), p. 1511-1516.

Young DM. (1971). Iterative Solution of Large Linear Systems. Academic Press, New York.

Youssef IK, Farid MM. (2015). On the accelerated overrelaxation method. *Pure and Applied Mathematics Journal*, 4(1), 26-31.

Youssef IK, Taha AA. (2013). On the modified successive overrelaxation method. *Applied Mathematics and Computation*, 219(9), 4601-4613.

Youssef IK. (2012). On the successive overrelaxation method. *Journal of Mathematics and Statistics*, 8, 176-184.