Path Planning of UAV based on Fluid Computing via Accelerated Method

Shaliza Hayati A. Wahab[1], Ali Chekima[2], Nordin Saad[3], Azali Saudi[4]

[1,3,4]Faculty of Computing and Informatics, Universiti Malaysia Sabah, Kota Kinabalu, Malaysia
[2]Faculty of Engineering, Universiti Malaysia Sabah, Kota Kinabalu, Malaysia
shaliza@ums.edu.my

ABSTRACT
This paper presents our study on the method for planning the path for low-flying unmanned aerial vehicle (UAV) in complex terrain based on the theory of fluid flow. First, the 2D terrain map is generated using hill algorithm. Then the fluid field distribution is computed using fluid mechanics to establish streamlines in the field. These streamlines are then used as flight paths for UAV. In fluid mechanics, Laplace's equation is used as the controlling equation of the potential flow. The solutions of Laplace's equation can be solved using finite difference method. Existing methods are slow in computing the solutions for generating UAVs flight path. Hence, they are not suitable for real-time processing. In this paper, an efficient iterative method namely Two-Parameter Overrelaxation (TOR) method that employ an acceleration parameter is proposed. The efficiency of this approach is shown by comparing its performance with the previous methods. It was shown that the proposed TOR method outperformed the previous methods.

Keywords: *UAV path planning, accelerated (AOR) iterative method, harmonic functions, Laplace's equation.*

I. INTRODUCTION

There has been a phenomenal growth in the world market as well as research in UAVs (Unmanned Aerial Vehicles) driven mainly by its wide-spread potential in civil applications used by general public. UAVs are unmanned flying machines capable of carrying out autonomous mission that can fly without a human pilot aboard [1]. The term drone is generically used, particularly by the media, as a description of all types of unmanned aircraft. UAVs have been used by the world's armed forces for wartime operations for more than 60 years for battlefield observations, and more recently in civil applications such as fire control and other kinds of surveillance, postal services and agriculture and wildlife to name few [2]. UAVs should be able to maneuver in complex terrain, plan path of their own while avoiding obstacles. To generate a suitable path for UAV, path planning should consider not only the potential impact of terrain on flight safety but also the performance constraints on UAV. Therefore, path planning needs to be done quickly especially for real-time processing.

The remainder of the paper is organized as follows. Section II describes the related works. In Section III, the background of harmonic functions in fluid computing is discussed. In Section VI, the harmonic functions in the SOR, AOR and TOR methods are derived. Results and relational graphs for TOR and AOR is presented in Section V. Later, in Section VI a numerical example is presented to illustrate the efficiency of the proposed method. Finally, Section VII draws the conclusions and future work is study**.**

II. RELATED WORKS

Over the years, the research regarding fast iterative schemes from the Successive Overrelaxation (SOR) and Accelerated Overrelaxation (AOR) family has initiated interest of researchers to in order to produce iterative methods that have faster convergence, lesser number of iteration and applicable than any methods [3]. Furthermore, the extended version of AOR family known as TOR method was reported in [11].

In [8], it was shown that the application of SOR for computing the harmonic functions had drastically improved the overall performance of the path planning algorithm. Recently, an extension to SOR namely Accelerated Overrelaxation (AOR) method was applied for navigation in various cases, for examples for mobile robot in indoor, static and structured environment [4]. Very recently, AOR method was also applied in image blending [5].

The purpose of this paper is to present the application of accelerated techniques, namely the AOR and TOR iterative methods that employ one and two acceleration parameters, respectively. The implementations of these iterative methods are then used for computing the harmonic functions in the field. Consequently, the computed harmonic functions will be used to find flight path for UAV in the simulated environment of 2D terrain map generated using hill algorithm.

III. HARMONIC FUNCTIONS IN FLUID COMPUTING

In recent years, fluid computation [6] or stream function [3],[10] is gaining more attention because of the planned smooth paths. In fluid flow method, the running water phenomenon can avoid rocks and flow to the terminal, thus a smooth flyable air path could be generated. The biggest advantage of this method is its high computational efficiency and smooth planned paths. This method imitates the phenomenon that water in river avoids rocks smoothly and reaches the destination eventually as shown in Fig. 1. As the streamlines obtained by simple formula still have certain optimizing properties, they can be available as planned paths for UAV.

Fig. 1. Fluid flow or streams

IV. HARMONICS FUNCTIONS

Harmonic function satisfies Laplace's equation

$$\nabla^2 u = \sum_{i=1}^{n} \frac{\partial^2 u}{\partial x_i^2} \quad (1)$$

Term x_i is the *i*-th Cartesian coordinate and *n* is the dimension. In the case of UAV path construction, the boundary of consists of all obstacles and goals in a configuration space representation. Harmonic functions satisfy the min-max principle [9], therefore spontaneous creation of a false local minimum inside the region is avoided if Laplace's equation is imposed as a constraint on the functions used. Numerical solutions for Laplace's equation are readily obtained from finite difference methods.

Although this system can be solved using direct method, the more efficient iterative methods are used to compute the solutions, since its application in path planning problem often resulting in large linear system with sparse coefficient matrix. The main advantage of iterative solution is that the storing of large matrices is unnecessary. However, one of the disadvantages of iterative methods compared with direct methods is slow convergence or even divergence. Thus, iterative method in practice requires an appropriate stopping criterion. The numerical solution of (1) can be solved effectively using SOR method. In [7], Hadjidimos introduced the AOR method, which is a two-parameter generalization of the SOR method. It was shown that in certain cases, the AOR method has better convergence rate than the classical Jacobi, Gauss-Seidel, or SOR method. Later, an extension of AOR method known as the Two-Parameter Overrelaxation (TOR) method was examined and produced encouraging results.

To solve path planning problem for UAV, Eq. (1) is rewritten in two-dimensional space as follows:

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0. \quad (2)$$

The finite difference approximation equation for Eq. (2) is defined as

$$u_{i,j} = \frac{1}{4}(u_{i-1,j} + u_{i+1,j} + u_{i,j-1} + u_{i,j+1}) \quad (3)$$

Eq. (3) is used to implement Gauss-Seidel (GS) iterative method and its computational molecule is shown in Fig. 2.

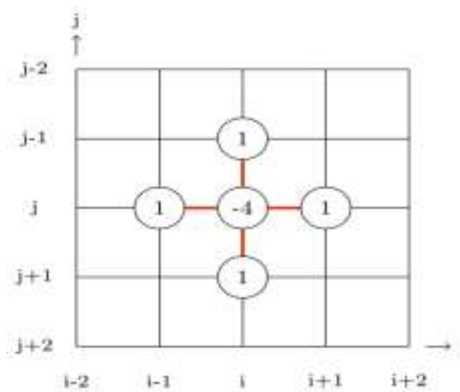


FIG. 2. THE COMPUTATIONAL MOLECULE OF EQUATION (3).

Based on Eq. (3), GS iterative scheme can be written as

$$u_{i,j}^{(k+1)} = \frac{1}{4}(u_{i-1,j}^{(k+1)} + u_{i+1,j}^{(k)} + u_{i,j-1}^{(k+1)} + u_{i,j+1}^{(k)}). \quad (4)$$

A. The Successive Overrelaxation (SOR) Method

The SOR employs a weighted parameter, ω, to the Gausss-Seidel iteration and is given below:

$$u_{i,j}^{(k+1)} = \frac{\omega}{4}\left(u_{i-1,j}^{(k+1)} + u_{i+1,j}^{(k)} + u_{i,j-1}^{(k+1)} + u_{i,j+1}^{(k)}\right) + (1-\omega)u_{i,j}^{(k)} \quad (5)$$

The extension of the SOR method is the Accelerated Overrelaxation (AOR) method for a linear system [7]. Further details of AOR method is described in Section B.

B. The Accelerated Overrelaxation (AOR) Method

The AOR employs an accelerated parameter, r, to the SOR iteration. Its iterative scheme is given below [7]:

$$u_{i,j}^{(k+1)} = \frac{\omega}{4}\left(u_{i-1,j}^{(k)} + u_{i+1,j}^{(k)} + u_{i,j-1}^{(k)} + u_{i,j+1}^{(k)}\right) + (1-\omega)u_{i,j}^{(k)} + \frac{r}{4}\left(u_{i-1,j}^{(k+1)} - u_{i-1,j}^{(k)} + u_{i,j-1}^{(k+1)} - u_{i,j-1}^{(k)}\right). \quad (6)$$

C. Two-Parameter Overrelaxation (TOR) Method

The TOR employs two accelerated parameters, r and s, to the AOR iteration. Its iterative scheme is given below [11]:

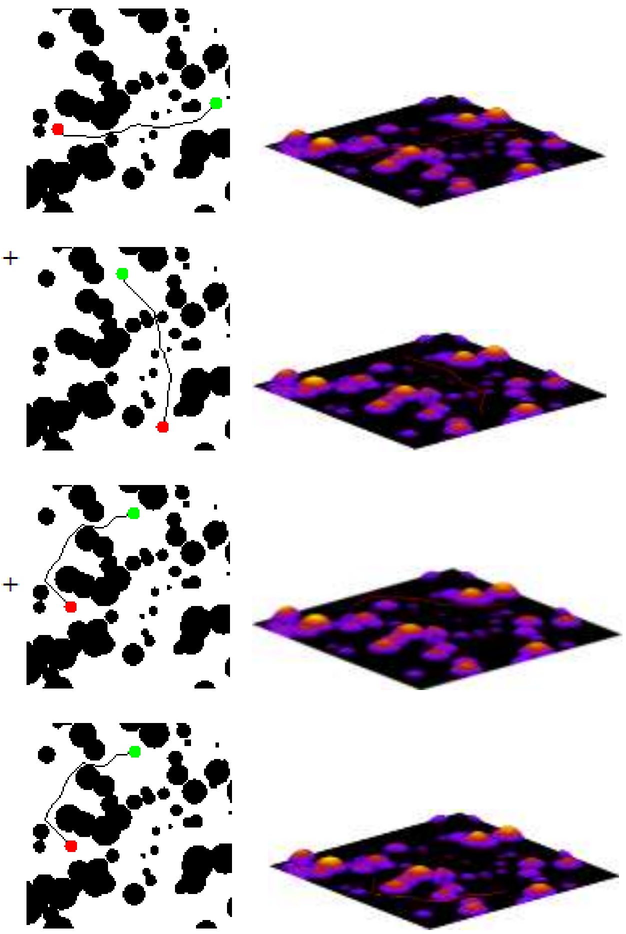
$$u_{i,j}^{(k+1)} = \frac{\omega}{4}\left(u_{i-1,j}^{(k)} + u_{i+1,j}^{(k)} + u_{i,j-1}^{(k)} + u_{i,j+1}^{(k)}\right) + (1-\omega)u_{i,j}^{(k)} + \frac{r}{4}\left(u_{i-1,j}^{(k+1)} - u_{i-1,j}^{(k)}\right) + \frac{s}{4}\left(u_{i,j-1}^{(k+1)} - u_{i,j-1}^{(k)}\right) \quad (7).$$

V. EXPERIMENTS AND RESULTS

A. Environment

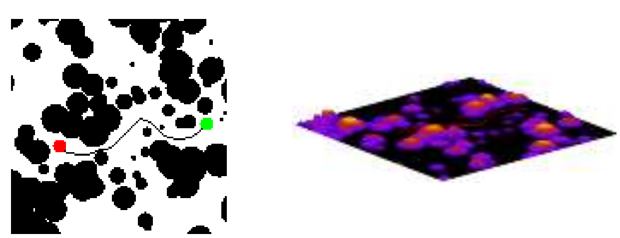
In order to test the robustness of the flight path planning algorithm, several experiments were carried out with various sizes of static environments, i.e. 128x128, 256x256, 512x512 and 1024x1024 that consists of a start and destination points. Fig. 3 shows the area covering 128x128 in random samples of 2D and followed by their corresponding 3D formats for 80 hills meanwhile Fig. 4 shows the area covering 128x128 in 2D and followed by their corresponding 3D formats for 120 hills. The start and destination points are denoted in red and green color, respectively. The dark circles in 2D and reddish purple in 3D represent obstacles. The UAV is supposed to avoid these obstacles while navigating the hilly terrain. These obstacles in real settings can either be buildings, trees, no fly zone, etc. and in our case we present hills as obstacles. This terrain map is generated using hill algorithm.

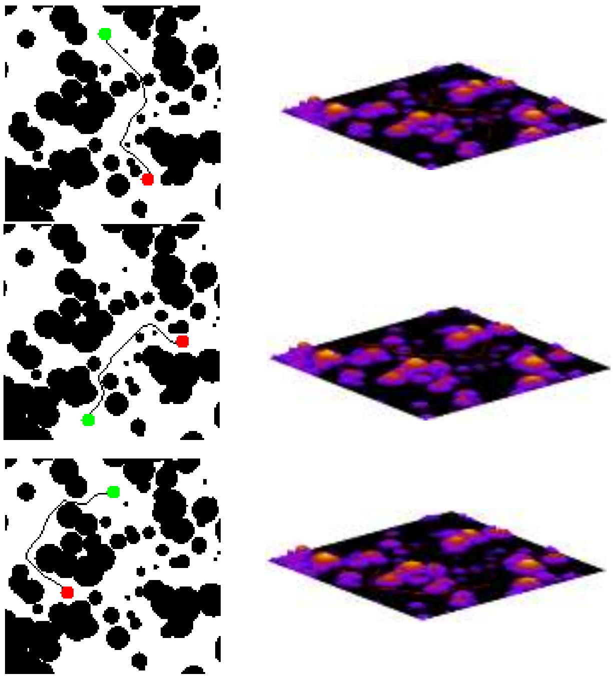
The flight path planning algorithm for the UAV begins by computing the harmonic functions that model the virtual fluid flow in the field. Three iterative methods, namely SOR, AOR and TOR, are considered to solve problem (1) to obtain the harmonic functions. The obtained harmonic functions create streamlines that flow from high point to the lowest point. By descending the generated streamlines from the harmonic functions, the goal point, which is the lowest point in the field, can be found. [8], [9].



(a) (b)

Fig. 3 (a). The generated random samples of 2d and (b) 3d paths for an area of 128 x 128 and 80 hills





(a) *(b)*

Fig. 4 (a). The generated random samples of 2d and (b) 3d paths for an area of 128 x 128 and 120 hills

B. Experimental Design

The generation of flight paths was simulated in the self-developed software written in Java, namely UAV Planner. The simulation was performed on computer with Windows 10 machine running on Intel i7 CPU at 1.8GHz speed equipped with 8GB of RAM. The parameters for the experimentation taken are defined as Table I.

Table I experiment parameters

Parameters	Details
Area coverage	128x128, 256x256, 512x512, 1024x1024
Number of hills	80, 120
Iteration Methods	SOR, AOR, TOR

C. RESULTS

The performance in terms of number of iterations for the tested methods are shown in this section. For sor method, the tested weighted parameter was in the range $1 \leq \omega < 2.0$. Whereas for aor method the tested weighted parameter for both ω and α were in the range $1 \leq \omega, \alpha < 2.0$. The tested weighted parameters for tor method were also in the range $1 \leq \omega, \alpha, s < 2.0$.

Table II Number of iterations for120 hills

METHODS	*Case 1 (128 x 128)*	*Case 2 (256 x 256)*	*Case 3 (512 x 512)*	*Case 4 (1024 x 1024)*
SOR	*230*	*1586*	*7280*	*27210*
AOR	*210*	*1257*	*5864*	*22443*
TOR	*217*	*1172*	*5503*	*21214*

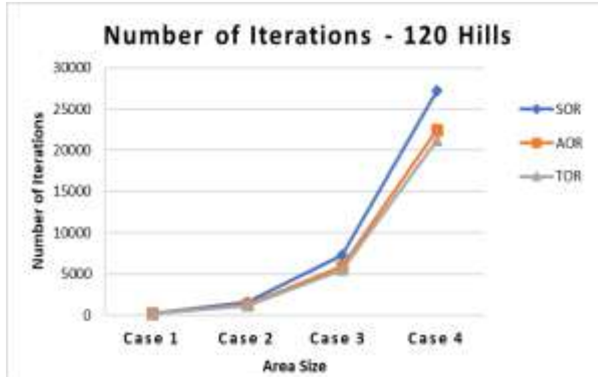


Fig. 5. Number Of Iterations Vs. Methods In Four Area For 120 Hills

Table II and Fig. 5 shows the results where number of hills is 120. The results show that AOR iterations perform faster than SOR. As for TOR, the iterations is generally slightly higher than AOR in 128x128 area. The number of iterations has decreased approximately 8.7 – 17% for AOR as compared to SOR. Meanwhile the number of iterations has reduced by approximately 3.3 – 5.5% for SOR while compared to AOR iterative methods. As the terrain size becomes bigger, the number of iterations using the TOR still outperforms the other methods.

Table III Number of Iterations for 80 Hills

METHODS	*Case 1 (128 x 128)*	*Case 2 (256 x 256)*	*Case 3 (512 x 512)*	*Case 4 (1024 x 1024)*
SOR	*290*	*1686*	*7635*	*32299*
AOR	*212*	*1335*	*6138*	*26195*
TOR	*220*	*1247*	*5756*	*24655*

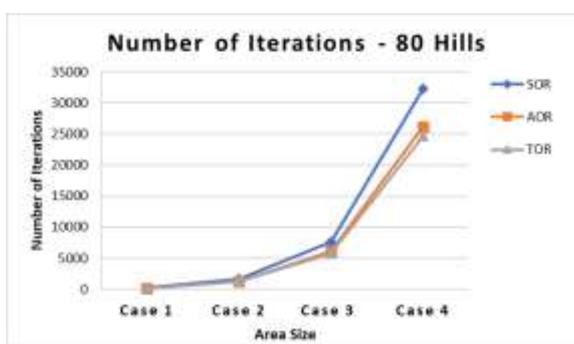


Fig. 6. Number of iterations vs. methods in four area for 80 hills

Table III and Fig. 6 shows the results where number of hills is 80. The results show that AOR iterations perform faster than SOR. As for TOR, the iterations is generally slightly hinger than AOR in 128x128 area. The number of iterations has decreased approximately 19 - 27 % for AOR as compared to SOR. Meanwhile the number of iterations of TOR has reduced by approximately 5.9% - 6.6% while compared with AOR iterative methods. It increase slightly only for 128x128 terrain. As the terrain size becomes bigger, the number of iterations using the TOR still outperforms the other methods.

Table VI Computational Time (ms) for No. of hills = 80 vs. 120

METHODS	*Hi L ls*	*Case 1* *(128 x 128)*	*Case 2* *(256 x 256)*	*Case 3* *(512 x 512)*	*Case 4* *(1024 x 1024)*
SOR	80	0 min, 0 sec, 176 ms	0 min, 1 sec, 575 ms	0 min, 23 sec, 57 ms	5 min, 1 sec, 816 ms
	120	0 min, 0 sec, 31 ms	0 min, 1 sec, 591 ms	0 min, 17 sec, 327 ms	5 min, 34 sec, 332 ms
AOR	80	0 min, 0 sec, 68 ms	0 min, 1 sec, 253 ms	0 min, 36 sec, 390 ms	4 min, 4 sec, 25 ms
	120	0 min, 0 sec, 26 ms	0 min, 1 sec, 150 ms	0 min, 13 sec, 702 ms	4 min, 58 sec, 663 ms
TOR	80	0 min, 0 sec, 119 ms	0 min, 1 sec, 20 ms	0 min, 25 sec, 720 ms	3 min, 26 sec, 186 ms
	120	0 min, 0 sec, 26 ms	0 min, 1 sec, 14 ms	0 min, 1 sec, 14 ms	3 min, 40 sec, 324 ms

Table IV shows the performance of tested methods in terms of computational time measured in milliseconds (ms) for different number of hills. AOR method clearly requires lesser computational time than SOR method. AOR and TOR shows more computational time in the less crowded area (80 hills) as it needs more resources in computing empty area. Overall, TOR method gave the best performance among the three tested methods.

VI. CONCLUSIONS AND FUTURE WORK

The effectiveness of fluid computing using AOR and TOR methods were demonstrated in this study, where both methods had contributed significantly in reducing the execution time of flight path planning algorithm for UAV. As an extension of this work, half-sweep iteration with AOR and TOR are interesting idea to explore.

ACKNOWLEDGEMENT

The work is supported by the Universiti Malaysia Sabah.

REFERENCES

[1] Y. Lu, Z. Xue, G. S. Xia, and L. Zhang, "*A survey on vision-based UAV navigation*," Geo-Spatial Inf. Sci., vol. 21, no. 1, pp. 21–32, 2018.

[2] F. Samaniego, J. Sanchis, S. García-Nieto, and R. Simarro, "*Recursive rewarding modified adaptive cell decomposition (RR-MACD): A dynamic path planning algorithm for UAVs*," Electron., vol. 8, no. 3, 2019.

[3] N. H. Mohd Ali and F. K. Pin, "*Modified explicit group AOR methods in the solution of elliptic equations*," Appl. Math. Sci., vol. 6, no. 49–52, pp. 2465–2480, 2012.

[4] A. Dahalan, A. Saudi, J. Sulaiman, and W. R. W. Din, "*Numerical evaluation of mobile robot navigation in static indoor environment via EGAOR Iteration*," J. Phys. Conf. Ser., vol. 890, no. 1, 2017.

[5] J. H. Eng, A. Saudi, and J. Sulaiman, "*Performance Analysis of Four-Point EGAOR Iterative Method Applied to Poisson Image Blending Problem*," Malaysian Journal of Science Special Issue (1), pp. 55–66, 2019.

[6] W. Honglun, L. Wentao, Y. Peng, L. Xiao, L. Chang, "*Three-dimensional path planning for unmanned aerial vehicle based on interfered fluid dynamical system*", Chinese Journal of Aeronautics, Volume 28, Issue 1, pp 229-239, ISSN 1000-9361, 2015.

[7] Hadjidimos, "*Accelerated Overrelaxation Method,*" Math. Comput., vol. 32, no. 141, p. 149-157, 1978.

[8] I. Connolly, J. B. Burns, and R. Weiss. 1990. "*Path planning using Laplace's equation*,". Proceedings of the IEEE International Conference on Robotics and Automation: pp. 2102–2106, 1990.

[9] S. Garrido, L. Moreno, D. Blanco, and F. Martín Monar, "*Robotic motion using harmonic functions and finite elements*," J. Intell. Robot. Syst. Theory Appl., vol. 59, no. 1, pp. 57–73, 2010.

[10] S. Waydo and R. M. Murray, "*Vehicle motion planning using stream functions*," 2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422), Taipei, Taiwan, vol.2, pp. 2484-2491. 2003.

[11] M. Madalena Martins, D. J. Evans & M. E. Trigo, "*Convergence of the interval and point TOR method*", International Journal of Computer Mathematics, vol. 80, no. 10, pp. 1227-1241, 2003