Autonomous path planning through application of rotated two-parameter overrelaxation 9-point Laplacian iteration technique

W. K. Ling[1], A. A. Dahalan[2], A. Saudi[3]
[1,2]Department of Mathematics, Universiti Pertahanan Nasional Malaysia (UPNM), Malaysia
[3]Faculty of Computing and Informatics, Universiti Malaysia Sabah, Malaysia

ABSTRACT

Autonomous path navigation is one of the important studies in robotics since a robot's ability to navigate through an environment brings about many advancements with it. This paper proposes the iteration technique called half-sweep two parameter overrelaxation 9-point Laplacian (HSTOR-9P) to be applied on autonomous path navigation and aims to investigate its effectiveness in performing computation for path planning in an indoor static environment. This iteration technique is a harmonic function that solves the Laplace's equation where the modelling of the environment is based on. The harmonic function is an appropriate method to be used on autonomous path planning because it satisfies the min-max principle, therefore, avoiding the occurrence of local minima which traps robot's movements and it also offers complete path planning algorithm. Its performance is tested against its predecessor iteration technique. Results show that HSTOR-9P iteration technique enables path construction in a lower number of iterations, thus, it performs better than its predecessors.

Corresponding Author:

W. K. Ling
Department of Mathematics
Universiti Pertahanan Nasional Malaysia
57000, Kuala Lumpur, Malaysia
Email: 3191191@alfateh.upnm.edu.my, a.qilah@upnm.edu.my

1. INTRODUCTION

Throughout the years in robotics development, path planning for robots has been a crucial study due to its importance. In the past, researchers have tried various methods to make path navigation in robotics more feasible [1], [2]. Each of these methods has its own advantages and drawbacks, as the environments the robot needs to traverse in are never uniform in terms of their offer in challenges and circumstances. Recent studies such as Handayani *et al.* [3] who conducted path planning analysis using swarm robots, Abdulredah and Kadhim [4] who integrated the method known as extended Kalman filter (EKF) with simultaneous localisation and mapping (SLAM) algorithm, and Das *et al.* [5] who developed a path planning algorithm known as OperativeCriticalPointBug (OCPB). Constant research gives opportunity for new findings that are beneficial for improvement. Computational load in autonomous path planning is still the crucial issue [6], therefore, a reduction is a must if efficiency is the desired output. In this study, the numerical method called HSTOR-9P is employed to efficiently solve path planning in a static environment. This is an iterative method and is based on the finite difference approximation. Previously, it is shown that the two-parameter overrelaxation (TOR) method which used the standard 5-point Laplacian operator provides promising path planning performance [7]. Thus, by implementing Half-Sweep (HS) approach and the 9-point Laplacian operator with TOR, the results are expected to be more superior. The numerical analysis in this study is based

on the theory of steady-state heat transfer. In order to implement the steady-state heat transfer theory, the configuration space is modelled using Laplace's equation. On that account, to solve the Laplace's equation, harmonic functions will be employed due to its practicality in autonomous path planning [8]-[10]. Various approaches were used in the past to obtain the harmonic functions. They include Jacobian method, Gauss-Seidel method, and SOR [11]. The most common method to obtain harmonic functions is via numerical technique, in which the HSTOR-9P iteration method is one of it. The harmonics functions obtained will be used to solve the numerically analysed temperature distributions in the operating environment, and the values will be used to generate path for the autonomous robot via a method called gradient descent search (GDS) method. The efficiency of the HSTOR-9P method will be compared to its predecessor iteration methods, i.e. successive overrelaxation 5-point Laplacian (SOR-5P), accelerated overrelaxation 5-point Laplacian (AOR-5P), two-parameter overrelaxation 5-point Laplacian (TOR-5P), successive overrelaxation 9-point Laplacian (SOR-9P), accelerated overrelaxation 9-point Laplacian (AOR-9P), two-parameter overrelaxation 9-point Laplacian (TOR-9P), Half-Sweep successive overrelaxation 5-point Laplacian (HSSOR-5P), Half-Sweep accelerated overrelaxation 5-point Laplacian (HSAOR-5P), Half-Sweep two-parameter overrelaxation 5-point Laplacian (HSTOR-5P), Half-Sweep successive overrelaxation 9-point Laplacian (HSSOR-9P), and Half-Sweep accelerated overrelaxation 9-point Laplacian (HSAOR-9P).

A pioneering study conducted by Khatib [12] introduced the application of potential fields in solving the problem of robot path planning. In Khatib's work, the concept of artificial potential field (APF) is introduced in robot arm's operational space. Meanwhile, in the configuration space, obstacles pulsated repelling forces while the objective point put out attractive force. Although potential field seemed practicable in path planning, it does suffer from the unfortunate creation of spurious local minima which caused the mobile robot to fall into premature stable configuration before reaching its objective point. However, Koditschek [13] showed that this might not be the case, at least for certain types of domain. Nonetheless, the integration of APF with suitable methods such as the study in [14] could provide benefits for autonomous path planning improvement. Due to its attractive properties, harmonic functions are a popular choice in solving path planning problems [15]-[17]. Through harmonic functions, which are also solutions to Laplace's equation, no local minimum is generated during path planning computations [9], [18]. Some other notable works involved the application of harmonic functions in path planning including the work by Waydo and Murray [19] where they investigated the use of stream functions on motion of vehicles. Other than that, Szulczynski *et al.* [20] had also applied harmonic functions in their study on its application on real-time obstacle avoidance.

By using finite elements, Garrido *et al.* [9] calculated harmonic functions for robotics locomotion. Apart from that, Daily and Bevly [21] applied analytical potential functions for solving complex shaped obstacles. The Half-Sweep (HS) iteration methods were introduced by Abdullah [22] where it was used to solve 2D Poisson equations. Saudi and Sulaiman [23] introduced the application of HS iteration method in robotics path planning and the method has shown promising results since [24], [25]. Standard iteration would employ 5-point Laplacian. However, the 9-point Laplacian approach is used in this paper to generate path for mobile robot. Saudi and Sulaiman [26] pioneered the application of 9-point Laplacian on mobile robot path planning. Adam *et al.* [27] observed that for a smooth initial estimation, 9-point Laplacian is expected to have more efficient convergence rate with more accurate computation.

2.   THE HEAT TRANSFER CONCEPT AND MODELLING OF ENVIRONMENTS

The environment models in this paper are constructed accordingly to treat the path planning of mobile robot as a steady-state heat transfer problem. By solving the Laplace's equation modelled after this heat transfer problem, we will be able to get the temperature values that represent the potential values that will be used by the mobile robot to build a smooth and unobstructed path for it to travel on. For this to work, the environment model will be set to such that heat will flow from a region of higher temperature to region of lower temperature. The environment model, from hereon to be called the configuration space, contains within it the exterior and interior boundary walls, a few obstacles, a starting point, and an objective point for which the mobile robot will be heading to from the starting point. The mobile robot representation is reduced to a single point in the configuration space. Boundary values are defined by the Dirichlet boundary conditions. The outer and inner boundary walls, together with obstacles, are assigned highest potential values to to act as heat source, while the objective point is assigned the lowest potential point to act as a heat sink. This way, a path will be able to be produced by executing the steepest descent search, that is GDS on the potential values. GDS is a search that follows the downward slope of potential fields from high to low values and can be used to create an unobstructed line of passage from its starting point towards its destination.

3. HARMONIC FUNCTIONS

Harmonic functions will be employed to compute the distribution of temperatures in the configuration space. In mathematical terms, a harmonic function on the domain is a function that satisfies the Laplace's equation:

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

where x_i is the i-th Cartesian coordinate and n is the dimension. In this study, the domain Ω is made up of exterior and interior boundary walls, all of the obstacles present, the starting point, and the objective point. To solve (1), the numerical method HSTOR-9P will be employed, upon computation we can obtain the function values relative to each node. As for the boundaries in the formula, Dirichlet boundary conditions will be used to constrain the Laplace's equation.

4. THE FORMULATION OF HSTOR-9P

In FS iteration, the whole computational grid is taken into iterative computation, and the nodal functions are iterated from the nodal set of the first point to the next nodal set which is its neighbouring point and so on sequentially. For HS iteration, only half of the computational grid is iterated, and the nodal sets are computed from the first point and move on to the next nodal set which moves in a rotated manner, this means the iteration skips a point in every iteration cycle. The visualisations for the individual nodal set for FS and HS iteration are as shown in Figure 1.

To understand the 9-point Laplacian visually, we compare it with the standard 5-point Laplacian model and show the distinction between both methods in a portion of configuration space, as shown in Figure 2. In one iteration cycle, it is clear that the HS 9-point Laplacian computes more point, resulting in increased accuracy in computational results due to more data available in a cycle of iteration, thus, a more efficient convergence rate.

(a)　　　　　　　　　　　　　　　　(b)

Figure 1. The 9-point Laplacian nodal sets for; (a) full-sweep; (b) half-sweep respectively
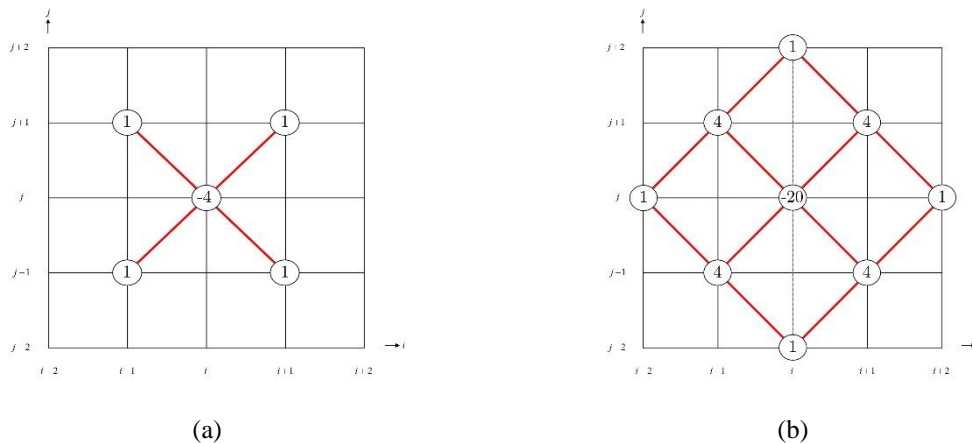

(a)　　　　　　　　　　　　　　　　(b)

Figure 2. The nodal function computational model (in bracket) in portion of configuration space for HS; (a) 5-point Laplacian; (b) 9-point Laplacian respectively

To put HSTOR-9P into formulation, from (1) the 2-dimensional Laplace's equation is given as:

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

To construct the standard HS 9-point Laplacian iteration formulation, (2) can be discretised based on computational model illustrated in Figure 2(b) and Figure 3(b), thus written as:

$$4(U_{i+1,j-1} + U_{i+1,j+1} + U_{i-1,j-1} + U_{i-1,j+1})$$
$$+ U_{i+2,j} + U_{i,j+2} + U_{i-2,j} + U_{i,j-2} - 20U_{i,j} = 0 \tag{3}$$

Implementing a weighted parameter, ω [28]-[30] into (3), the formula for HSSOR-9P can be defined as:

$$U_{i,j}^{k+1} = \frac{\omega}{5}(U_{i-1,j-1}^{k+1} + U_{i-1,j+1}^{k} + U_{i+1,j-1}^{k+1} + U_{i+1,j+1}^{k})$$
$$+ \frac{\omega}{20}(U_{i-2,j}^{k+1} + U_{i+2,j}^{k} + U_{i,j-2}^{k+1} + U_{i,j+2}^{k}) + (1-\omega)U_{i,j}^{k} \tag{4}$$

From (4), the method can be further enhanced to form the formula for an enriched iteration of the same overrelaxation iteration family, with HS 9-point Laplacian characteristics, namely HSAOR-9P. To build the formulation, a second weighted parameter, r, is extracted and implemented into (4). By replacing $U_{i-1,j-1}^{k+1}$, $U_{i+1,j-1}^{k+1}$, $U_{i-2,j}^{k+1}$ and $U_{i,j-2}^{k+1}$ with $U_{i-1,j-1}^{k}$, $U_{i+1,j-1}^{k}$, $U_{i-2,j}^{k}$ and $U_{i,j-2}^{k}$ respectively and adding the terms $\frac{r(U_{i-1,j-1}^{k+1} - U_{i-1,j-1}^{k})}{5}$, $\frac{r(U_{i+1,j-1}^{k+1} - U_{i+1,j-1}^{k})}{5}$, $\frac{r(U_{i-2,j}^{k+1} - U_{i-2,j}^{k})}{20}$, and $\frac{r(U_{i,j-2}^{k+1} - U_{i,j-2}^{k})}{20}$, we obtain the formula for HSAOR-9P as:

$$U_{i,j}^{k+1} = \frac{r}{5}\left(U_{i-1,j-1}^{k+1} - U_{i-1,j-1}^{k} + U_{i+1,j-1}^{k+1} - U_{i+1,j-1}^{k}\right)$$
$$+ \frac{r}{20}(U_{i-2,j}^{k+1} - U_{i-2,j}^{k} + U_{i,j-2}^{k+1} - U_{i,j-2}^{k})$$
$$+ \frac{\omega}{5}(U_{i-1,j-1}^{k} + U_{i-1,j+1}^{k} + U_{i+1,j-1}^{k} + U_{i+1,j+1}^{k}) \tag{5}$$
$$+ \frac{\omega}{20}(U_{i-2,j}^{k} + U_{i+2,j}^{k} + U_{i,j-2}^{k} + U_{i,j+2}^{k}) + (1-\omega)U_{i,j}^{k}$$

Finally, from (5), we improve the HSAOR-9P iteration by drawing out a third weighted parameter, s to obtain the formulation for HSTOR-9P. This is formulated by replacing $\frac{r(U_{i+1,j-1}^{k+1} - U_{i+1,j-1}^{k})}{5}$ and $\frac{r(U_{i,j-2}^{k+1} - U_{i,j-2}^{k})}{20}$ with $\frac{s(U_{i+1,j-1}^{k+1} - U_{i+1,j-1}^{k})}{5}$ and $\frac{s(U_{i,j-2}^{k+1} - U_{i,j-2}^{k})}{20}$ respectively. Thus, we obtain:

$$U_{i,j}^{k+1} = \frac{r}{5}(U_{i-1,j-1}^{k+1} - U_{i-1,j-1}^{k}) + \frac{s}{5}(U_{i+1,j-1}^{k+1} - U_{i+1,j-1}^{k})$$
$$+ \frac{r}{20}(U_{i-2,j}^{k+1} - U_{i-2,j}^{k}) + \frac{s}{20}(U_{i,j-2}^{k+1} - U_{i,j-2}^{k})$$
$$+ \frac{\omega}{5}(U_{i-1,j-1}^{k} + U_{i-1,j+1}^{k} + U_{i+1,j-1}^{k} + U_{i+1,j+1}^{k}) \tag{6}$$
$$+ \frac{\omega}{20}(U_{i-2,j}^{k} + U_{i+2,j}^{k} + U_{i,j-2}^{k} + U_{i,j+2}^{k}) + (1-\omega)U_{i,j}^{k}$$

By applying (6), a harmonic function that solves the Laplace's equation into computation, we obtain the potential values which will be used by the mobile robot to generate path through GDS technique. The value of ω is determined through sensitivity analysis and is set at $1.8 \le \omega \le 2.0$ and the values for weighted parameter r and s are also tested through sensitivity analysis and generally set to be as approximate but not the same as the value of ω [31]. To ensure that the computational results can avoid flat region in calculations, the iterations were set to stop when the maximum error of the solutions falls inside the range of specified tolerance error, at either 8.8818 E^{-16} or 9.9920 E^{-16}.

5.   EXPERIMENTS AND RESULTS

For the setting up of the experiment, 4 environments had been prepared based on environment models in [32] and were labelled as Environment 1, Environment 2, Environment 3, and Environment 4. Each configuration space used consists of different number of obstacles, from 2 obstacles to 5 obstacles. These setups were chosen to observe whether the density of obstacles present in the configuration space affects the iteration rate of the mobile robot. All environments contained the same basic setup, which consisted of the outer and inner boundary walls, obstacles together with the user defined starting point, and objective point. To accommodate the heat transfer theory, temperature values were distributed varyingly throughout the configuration space. The outer and boundary walls, together with obstacles, were appointed high temperature values, whereas the objective point was appointed the lowest temperature value, acting like a heat sink pulling all the heat in creating heat flux lines. Upon completion of iteration, the mobile robot will follow the heat flux line by searching via GDS technique and produce a smooth unobstructed path towards the objective point. The path produced by all the iterative methods are the same, as shown in Figure 3, the difference is in the number of iterations the iterative methods need to complete the path planning computation. The higher the number of iterations, the higher the computational load, regardless of the machine used for computation. For the simulation, the programme used was named Robot 2D Simulator [33], written on the Delphi software for efficient computation. Delphi is an efficient compiler that allows faster coding production. The computation was performed on a computer with Intel Core i5-5200U CPU at 2.2 GHz utilising 4 GB of RAM.
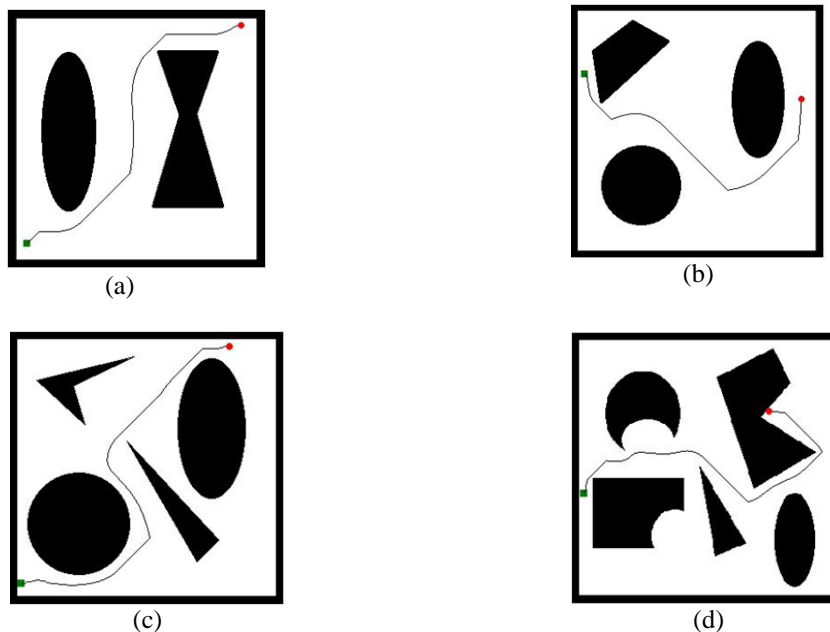

(a)

(b)

(c)

(d)

Figure 3. Path lines generated via Robot 2D Simulator using HSTOR-9P with details for; (a) Environment 1: Number of obstacles: 2, Free space: 67%, Occupied space: 33%; (b) Environment 2: Number of obstacles: 3, Free space: 69%, Occupied space: 31%; (c) Environment 3: Number of obstacles: 4, Free space: 63%, Occupied space: 37%; (d) Environment 4 respectively: Number of obstacles: 5, Free space: 62%, Occupied space: 38%. Green coloured rectangle points as starting point, red dot as objective point

The results are shown in Table 1 and Table 2. When compared to other iteration of the same overrelaxation family methods, HSTOR-9P clearly showed more efficient performance in terms of iteration number needed to complete computation. In terms of CPU time, it was observed that in smaller environment, i.e. 300×300, the difference in time was not too distinct, but became more definite in difference in larger sized environments. This is due to background processes in the computer that have effect on CPU processing. Thus, the effects rippled on smaller environment computation where CPU processing time frame is small (less than 5 seconds). It should also be noted that the number of iterations altogether is the highest for environment 2, while the lowest altogether in environment 4. With reference to Figure. 1, the cost of computation is the highest in the environment with the most percentage of free space area, and vice versa.

This means that the more space occupied by obstacles, regardless of the number, the less cost of computation is required. At a glance, it is clear that the HS approach is more superior in performance compared to the FS approach, and HSTOR-9P method requires the least overall number of iterations to complete the path planning computation. For future studies, it is proposed that the quarter-sweep iterative method [34]-[37] to be implemented for TOR iteration method for improvements in autonomous robot path planning.

Table 1. Performance of the examined numerical method based on number of iterations

	Methods	N x N		
		300 x 300	600 x 600	900 x 900
Environment 1	SOR-5P	1312	4185	7398
	AOR-5P	1151	3105	5417
	TOR-5P	979	2730	4350
	HSSOR-5P	1095	3050	4747
	HSAOR-5P	916	2562	3741
	HSTOR-5P	683	2060	3187
	SOR-9P	1231	3612	6541
	AOR-9P	1113	2805	4771
	TOR-9P	955	2356	3861
	HSSOR-9P	918	2581	4094
	HSAOR-9P	783	2191	3211
	HSTOR-9P	619	1736	2735
Environment 2	SOR-5P	2231	4868	10916
	AOR-5P	1726	3467	8110
	TOR-5P	1413	1987	5060
	HSSOR-5P	1241	3065	6968
	HSAOR-5P	914	2329	5513
	HSTOR-5P	834	1906	3961
	SOR-9P	1878	4272	9667
	AOR-9P	1549	3055	7159
	TOR-9P	1223	1802	4680
	HSSOR-9P	1040	2624	6034
	HSAOR-9P	848	1995	4778
	HSTOR-9P	723	1623	3444
Environment 3	SOR-5P	1544	5541	9034
	AOR-5P	1093	4081	7560
	TOR-5P	795	3318	5107
	HSSOR-5P	948	2362	7046
	HSAOR-5P	757	1964	5398
	HSTOR-5P	596	1056	3737
	SOR-9P	1284	4662	7722
	AOR-9P	908	3448	6457
	TOR-9P	687	2810	4405
	HSSOR-9P	834	1989	5909
	HSAOR-9P	661	1651	4568
	HSTOR-9P	561	919	2425
Environment 4	SOR-5P	766	2662	4962
	AOR-5P	748	2129	3884
	TOR-5P	687	1322	3294
	HSSOR-5P	567	1575	4160
	HSAOR-5P	524	1059	3057
	HSTOR-5P	460	903	2450
	SOR-9P	756	2253	4264
	AOR-9P	736	1798	3322
	TOR-9P	674	1165	2823
	HSSOR-9P	550	1275	3512
	HSAOR-9P	482	928	2574
	HSTOR-9P	455	793	2067

Table 2. Execution results of the examined numerical methods based on CPU time (in seconds)

	Methods	N x N		
		300 x 300	600 x 600	900 x 900
Environment 1	SOR-5P	1.86	35.00	168.77
	AOR-9P	1.86	29.05	137.24
	TOR-9P	1.62	25.75	111.90
	HSSOR-9P	0.95	18.16	72.53
	HSAOR-9P	0.88	15.20	60.66
	HSTOR-9P	0.66	12.83	53.46
	SOR-9P	1.99	32.99	168.01
	AOR-9P	2.28	31.47	146.46
	TOR-9P	1.98	26.20	124.90
	HSSOR-9P	0.88	16.39	67.72
	HSAOR-9P	0.91	15.53	62.64
	HSTOR-9P	0.75	12.52	54.10
Environment 2	SOR-5P	3.23	42.14	247.08
	AOR-5P	2.88	33.56	208.82
	TOR-5P	2.36	19.11	135.19
	HSSOR-5P	1.09	18.39	110.16
	HSAOR-5P	0.74	13.85	91.88
	HSTOR-5P	0.86	12.19	67.13
	SOR-9P	3.13	41.08	249.36
	AOR-9P	3.23	34.44	222.62
	TOR-9P	2.58	20.56	153.16
	HSSOR-9P	1.02	17.24	102.72
	HSAOR-9P	1.02	14.38	94.00
	HSTOR-9P	0.91	11.86	69.41
Environment 3	SOR-5P	2.11	47.33	197.27
	AOR-5P	1.70	36.49	185.62
	TOR-5P	1.23	30.38	125.91
	HSSOR-5P	0.77	13.50	104.06
	HSAOR-5P	0.69	11.55	82.93
	HSTOR-5P	0.56	6.42	59.74
	SOR-9P	1.98	41.28	190.95
	AOR-9P	1.76	36.08	188.60
	TOR-9P	1.37	30.06	132.55
	HSSOR-9P	0.78	12.30	92.22
	HSAOR-9P	0.73	11.27	82.57
	HSTOR-9P	0.66	6.37	44.81
Environment 4	SOR-5P	1.05	21.19	103.46
	AOR-5P	1.17	18.77	91.91
	TOR-5P	1.08	12.35	79.11
	HSSOR-5P	0.47	8.53	59.55
	HSAOR-5P	0.53	5.94	46.35
	HSTOR-5P	0.44	5.42	38.47
	SOR-9P	1.17	19.75	100.79
	AOR-9P	1.44	18.74	94.55
	TOR-9P	1.33	12.35	83.33
	HSSOR-9P	0.52	7.81	54.69
	HSAOR-9P	0.56	6.20	46.42
	HSTOR-9P	0.55	5.42	38.17

6. CONCLUSIONS AND FUTURE WORKS

The autonomous path planning performance of the integration of the standard TOR iterative method with HS approach and 9-point Laplacian operator. Its performance is compared against the predecessing methods; they are SOR-5P, AOR-5P, TOR-5P, SOR-9P, AOR-9P, TOR-9P, HSSOR-5P, HSAOR-5P, HSTOR-5P, HSSOR-9P, and HSAOR-9P. The path planning performance is measured via number of iterations an iterative method needs to complete the path planning computation. Based on the results obtained, it is shown that the path planning performance of HSTOR-9P supercedes its predecessors. Through the Robot 2D Simulator programme used to simulate the path planning, the autonomous robot was able to

generate a smooth path from the starting point to the objective point, while avoiding all the obstacles in between. Thus, it can also be concluded that by applying HS approach and 9-point Laplacian operator, the performance of autonomous robot can be improved compared to their standard predecessor approach, i.e. FS approach and 5-point Laplacian operator respectively.

REFERENCES

[1] H. Y. Zhang, W. Lin, and A. Chen, "Path Planning for The Mobile Robot: A Review," *Symmetry*, vol. 10, no. 10, pp. 450, 2018, doi: 10.3390/sym10100450.
[2] A. Koubaa *et al.*, "Introduction to Mobile Robot Path Planning," *Robot Path Planning and Cooperation*, pp. 3–12, 2018, doi: 10.1007/978-3-319-77042-0_1.
[3] A. S. Handayani, S. Nurmaini, I. Yani, and N. L. Husni, "Analysis on Swarm Robot Coordination using Fuzzy Logic," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 13, no. 1, , pp. 48-57, 2020, doi: 0.11591/ijeecs.v13.i1.pp48-57.
[4] S. H. Abdulredah and D. J. Kadhim, "Developing a Real Time Navigation for Mobile Robots at Unknown Environments," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 20, no. 1, pp. 500-509, 2020, doi: 10.11591/ijeecs.v20.i1.pp500-509.
[5] S. K. Das, A. K. Dutta, and S. K. Debnath, "OperativeCriticalPointBug Algorithm-Local Path Planning of Mobile Robot Avoiding Obstacles," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 18, no. 3, pp. 1646-1656, 2020, doi: 10.11591/ijeecs.v18.i3.pp1646-1656.
[6] M. N. Zafar, and J. C. Mohanta, "Methodology for Path Planning and Optimization of Mobile Robots: A Review," *Procedia Computer Science*, vol. 133, pp. 141–152, 2018, doi: 10.1016/j.procs.2018.07.018.
[7] S. H. A. Wahab, A. Chekima, N. Saad, and A. Saudi, "Path Planning of UAV based on Fluid Computing via Accelerated Method," *International Journal of Engineering Trends and Technology*, pp. 76-80, 2020. doi: 10.14445/22315381/CATI2P212
[8] C. I. Connolly and R. A. Gruppen, "The Applications of Harmonic Functions to Robotics," Journal of Robotic Systems, vol. 10, no. 7, pp. 931-946, 1993, doi: 10.1002/rob.4620100704.
[9] S. Garrido, Luis Moreno, Dolores Blanco, and Fernando Martín Monar, "Robotic Motion using Harmonic Functions and Finite Elements," *Journal of Intelligent and Robotic Systems*, vol. 59, pp. 57–73, 2010, doi: 10.1007/s10846-009-9381-3.
[10] S. Panati, Bayanjargal Baasandorj᾿ and Kil To Chong, "Autonomous Mobile Robot Navigation using Harmonic Potential Field," *IOP Conference Series: Materials Science and Engineering*, vol. 83, 2015, doi: 10.1088/1757-899X/83/1/012018.
[11] S. Sasaki, "A Practical Computational Technique for Mobile Robot Navigation," in *Proceedings of the 1998 IEEE International Conference on Control Applications*, vol. 2, 1998, pp. 1323-1327.
[12] O. Khatib, "Real-Time Obstacles Avoidance for Manipulators and Mobile Robot," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 2, no. 1, 1985, pp. 500-505, doi: 10.1007/978-1-4613-8997-2_29.
[13] D. E. Koditschek, "Exact Robot Navigation by Means of Potential Functions: Some Topological Considerations," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 4, 1987, pp. 1-6.
[14] E. Falomir, S. Chaumette and G. Guerrini, "A 3D Mobility Model for Autonomous Swarms of Collaborative UAVs," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2019, pp. 196-204.
[15] A. Masoud and A. Al-Shaikhi, "Sensor-Based, Time-Critical Mobility of Autonomous Robots in Cluttered Spaces: A Harmonic Potential Approach," *Robotica*, vol. 36, no. 9, pp. 1301-1332, 2018, doi: 10.1017/S0263574718000401.
[16] A. Falcó, L. Hilario, N. Montés, M, C. Mora, and E. Nadal, "A Path Planning Algorithm for a Dynamic Environment Based on Proper Generalized Decomposition," *Mathematics*, vol. 8, no. 12, 2020, doi: doi.org/10.3390/math8122245.
[17] A. Saudi and J. Sulaiman, "Harmonic Path Planning using Two-Stage Half-Sweep Arithmetic Mean Method," *Journal of Computer Science & Computational Mathematics*, vol. 9, no. 2, pp. 23-27, 2019, doi: 10.20967/JCSCM.2019.02.002.
[18] C. I. Connolly, J. B. Burns, and R. Weiss., "Path Searching using Laplace's Equation," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 3, 1990, pp. 2102-2106, doi: 10.1109/ROBOT.1990.126315.
[19] S. Waydo and R. M. Murray, "Vehicle Motion Planning using Stream Functions," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 2, 2003, pp. 2484-2491, doi: 10.1109/ROBOT.2003.1241966.

[20] Szulczyński P., Pazderski D., Kozłowski K., "Real-Time Obstacle Avoidance using Harmonic Potential Functions," *Journal of Automation, Mobile Robotics and Intelligent Systems*, vol. 5, no. 3, pp. 59-66, 2011.

[21] R. Daily and D. M. Bevly, "Harmonic Potential Field Path Searching for High Speed Vehicles," in *Proceedings of IEEE American Control Conference*, 2008, pp. 4609-4614, doi: 10.1109/ACC.2008.4587222.

[22] A. R. Abdullah, "The Four-Point Explicit Decoupled Group (EDG) Method: A Fast Poisson Solver," *International Journal of Computer Mathematics*, vol. 38, no. 1-2, pp. 61–70, 1991, doi: 10.1080/00207169108803958.

[23] A. Saudi and J. Sulaiman, "Half-Sweep Gauss-Seidel (HSGS) Iterative Method for Robot Path Planning," in *The 3rd International Conference on Informatics and Technology (Informatics09)*, 2009, pp. 34-39.

[24] A. Saudi and J. Sulaiman, "Path Planning for Mobile Robot With Half-Sweep Successive Over-Relaxation (HSSOR) Iterative Method," in *Proceedings of the Symposium on Progress in Information & Communication Technology (SPICT '09)*, 2009, pp. 57-62.

[25] A. Saudi and J. Sulaiman, "Red-Black Strategy for Mobile Robot Path Planning," in *Proceedings of the International Multi Conference of Engineers and Computer Scientists, IMECS'10; Topics: The IAENG International Conference on Scientific Computing, ICSC'10*, vol. 3, 2010, pp. 2215-2219.

[26] A. Saudi J. Sulaiman, "Block Iterative Method using Nine-Point Laplacian for Robot Path Planning," *European Journal of Scientific Research*, vol. 43, no.2, pp. 204-211, 2010.

[27] L. M. Adams, R. J. LeVeque, and D. M. Young, "Analysis of the SOR Iteration for the 9-Point Laplacian," *SIAM Journal on Numerical Analysis*, vol. 25, no. 5, pp. 1156-1180, 1988, doi: 10.1137/0725066.

[28] D. M. Young, *Iterative Solution of Large Linear Systems*, London: Academic Press, 1971.

[29] D. M. Young, "Second-Degree Iterative Methods for the Solution of Large Linear Systems," *Journal of Approximation Theory*, vol. 5, no.2, pp. 37-148, 1972, doi: 10.1016/0021-9045(72)90036-6.

[30] D. M. Young, "Iterative Solution of Linear Systems Arising from Finite Element Techniques," Whiteman, J.R.. *The Mathematics of Finite Elements and Applications*, *London: Academic Press*, vol. 2, pp. 439-464, 1976.

[31] A. Hadjidimos, "Accelerated Overrelaxation Method," *Mathematics of Computation*, vol. 32, no. 141, pp. 149-157, 1978, doi: 10.1090/S0025-5718-1978-0483340-6.

[32] A. Zelinsky, "Environment Exploration and Path Searching Algorithms for Mobile Robot Navigation Using Sonar," Doctoral dissertation, Department of Computer Science, University of Wollongong, Northfields Ave, Australia, 1991.

[33] A. Saudi, "Robot Path Planning using Family of SOR Iterative Methods with Laplacian Behaviour-Based Control," Ph.D thesis, Universiti Malaysia Sabah, Malaysia, 2015.

[34] A. A. Dahalan, M. S. Muthuvalu, and J. Sulaiman, "QSAGE Iterative Method Applied to Fuzzy Parabolic Equation," in *AIP Conference Proceedings*, 2016, vol. 1705, no. 1, doi: 10.1063/1.4940272.

[35] A. A. Dahalan, N. S. A. Aziz, and J. Sulaiman, "Performance of Quarter-Sweep Successive Overrelaxation iterative method for Two-Point Fuzzy Boundary Value Problems," *Journal of Engineering and Applied Sciences*, vol. 11, no. 7, pp. 1456-1463, 2016, doi: 10.3923/jeasci.2016.1456.1463.

[36] A. A. Dahalan, N. S. A. Aziz, and J. Sulaiman, "QSAGE Iterative Method for Numerical Solution of Two-Point Fuzzy Boundary Value Problem," *Journal of Engineering and Applied Sciences*, vol. 12, no. 16, pp. 4068-4074, 2017, doi: 10.36478/jeasci.2017.4068.4074.

[37] A. A. Dahalan, A. Saudi, and J. Sulaiman, "Numerical Technique for Autonomous Robot Path Searching Based On QSAOR Iterative Method using Indoor Environment," *Journal of Engineering and Applied Sciences*, vol. 13, no. 20, pp. 8414-8418, 2018, doi: 10.36478/jeasci.2018.8414.