A New Gap Selection Strategy for Follow the Gap Method

Hosein Houshyari and Volkan Sezer
Autonomous Mobility Group, Control & Automation Engineering Department, Istanbul Technical University, Istanbul, Turkey
Email: houshyari18@itu.edu.tr, sezerv@itu.edu.tr

***Abstract*— Obstacle avoidance methods guarantee the robot's safety during the tracking of the planned path. Follow the Gap Method known (FGM) is a geometry-based obstacle avoidance method that continuously leads the robot to the goal point by selecting the largest gap existing around the robot. This approach calculates the heading angle by considering the distance to the closest obstacles, the angle to the goal, and the center of the gap. In this paper, a new procedure is developed to improve the gap selection in FGM, where the gaps are selected based on the prediction of gap changes during the time, considering the distance between the robot and obstacles in the future. In order to test the proposed methodology, Monte-Carlo simulations are used and the results are presented for comparison. The results demonstrate that the new procedure leads the robot to safer trajectories in comparison with classical FGM.**

***Index Terms*— obstacle avoidance, autonomous robots, path planning, dynamic obstacle**

I. Introduction

The ability of robots to travel from the initial state toward the final state without any collisions using the sensorial data and environmental information is performed by path planning. Path planning algorithms are divided into two major classes: Global and local/reactive path planning [1], [2]. When the obstacles are solid and static, and the map of the environment is not changing and being updated dynamically, the global planning algorithms can calculate the travel trajectory on the collision-free spaces of the map even before the travel. In this scenario, the robot is able to reach the final state by just following the produced path. For path planning purposes, different methodologies like A* [3], Dijkstra [4], Rapidly Exploring Random Trees (RRTs) [5], RRT* [6], and cell decomposition methods [7] have been developed. However, the uncertainty and unavailability of environmental information can challenge the accuracy of such approaches in pathfinding processes.

Local path planning algorithms can be useful when the environment is unknown and it contains dynamic obstacles or the motion of the robot contains uncertainty. In this class of path planning methods, the main goal is to avoid the robot from colliding with the detected obstacles while moving toward the goal state. In recent years, there has been great improvement in developing local path planning algorithms.

The earliest method in this category is the Bug algorithm, presented in [8]. In this algorithm, the robot moves directly toward the goal state and when facing an obstacle, the robot starts to moves around it until there is no obstacle to the goal state and then continues moving toward the goal. One of the other common obstacle avoidance methods is Artificial Potential Field (APF) [9]. Using the potential field concepts, the obstacles and the goal of this algorithm are respectively represented by repulsive artificial and attractive potentials. The attractive potential field generates a pulling force towards the goal and the repulsive potential fields coming from the obstacles generate pushing forces to push the robot away. Using these forces, the robot gets the ability to avoid obstacles while moving towards the goal state. Regardless of its simplicity, the main disadvantage of the APF method is when facing a local minimum [10]. The local minimum happens when the forces on the robot cancel each other, leading the robot to get stuck. To fix these issues in the APF method, appropriate solutions are presented in [11], [12]. Another method used for obstacle avoidance is the Virtual Force Field method (VFF) [13]. In this method, a two-dimensional Cartesian histogram grid is used to demonstrate the obstacles. Each cell in the histogram grid represents the chance of including an obstacle at that specific point. Then, the APF is applied to the histogram grid and consequently, the local minima problem remains in the VFF method. In another methodology called Vector Field Histogram (VFH) approach [14], the two-dimensional Cartesian histogram grid at the current location of the robot is reduced to a one-dimensional polar histogram. Later, VFH selects the field with the lowest polar obstacle density among all other polar fields and sets the heading angle of the robot according to the direction resulted from the chosen field. The methods discussed above are further improved in some other studies in the literature [15]. Dynamic Window Approach (DWA) [16] is one of the popular obstacle avoidance algorithms that considers the acceptable velocity sets for the robot. These acceptable velocity sets are appraised by maximizing an objective function. There are also some studies to improve the performance of DWA like [17].

Manuscript received November 4, 2020; revised April 21, 2021.

doi: 10.18178/ijmerr.10.5.248-254

Follow the Gap Method known as (FGM) [18], is a geometrical-obstacle avoidance method that cares most about the safety. The FGM continuously guides the robot to the goal point by selecting the largest available gap around the robot. It calculates the heading angle by considering the distance to the closest obstacles and the angle to the goal and the center of the gap. Multiple improvements have been made for this method in the literature. Some of these methods try to fix the available issues like the Zigzag behavior and the trajectory length problem of the classic FGM like Improved Follow the Gap Method (FGM-I) [19]. Another version is called FGM-DWA [20] which uses the FGM's safe heading angle and calculates the admissible velocities for the robot using the DWA algorithm. An alternative application of FGM for the overtaking maneuver is provided in [21]. The latest work which is called FGM-I2 combines FGM-I with DWA as shown in [22].

In this paper, a novel procedure is presented for the gap selection step of the FGM methodology. The classic FGM selects the largest gap and no prediction is available for the gap changes. Additionally, this method only considers the existence of the obstacles using sensory data, and no other information about the obstacles is taken into account in this method. In the proposed novel FGM, follow the dynamic gap method (FDGM), the gaps are selected based on the prediction of gap changes during the time, considering a possible collision between the robot and the obstacles.

The rest of the paper is structured as follows: Section II introduces the FGM. Section III explains the new procedure for the gap selection step of FGM. Simulation results are presented in section IV and the conclusions are presented in Section V.

II. Follow the Gap Method

FGM is a safety-geometry based local obstacle avoidance algorithm, because of its inherent characteristics FGM leads the robot to the safer trajectories, [19]. And the efficiency of this approach is validated in different studies [23]. Before the implementation of FGM, the robot is assumed to be circular and the obstacles are inflated with the robot's radius so that the robot can be considered as a point for the rest of the algorithm Implementing FGM consists of three main steps: 1) FGM calculates the gap arrays from the polar obstacle distance arrays coming from the sensory information and it finds the largest gap (gap selection step). 2) FGM calculates the gap center angle 3) the final heading angle calculation is performed concerning a safety factor named alfa, largest gap center angle, minimum distance to obstacles, and the goal angle using the Equation (1).

$$\varphi_{final} = \frac{\frac{\alpha}{dmin}\,\varphi_{gap-c} + \varphi_{goal}}{\frac{\alpha}{dmin} + 1} \quad (1)$$

The Eq. (1) clearly shows that by getting the obstacles closer to the robot (dmin → 0), FGM considers safety. The alfa coefficient shows the state of keeping a distance from the obstacles, which means that for the higher values of alfa, the robot gets distance from the obstacles and selects the center of the safe gap. On the other hand, lower values of alfa cause the robot not to afraid of the obstacles and make it move closer to the obstacles. Fig. 1. represents the robot-obstacle configuration, the gaps, midpoint of the largest gaps, the goal point, and the final heading angle.

III. Follow the Dynamic Gap Method

Although FGM has proved itself as one of the safest methods in obstacle avoidance, it does not use any smart procedure to select the gaps. This inherent characteristic of FGM causes it to be unsafe in some robot-obstacle configurations as illustrated in Fig. 2. In Fig. 2, it is seen that there is a dynamic obstacle moving upside down. The path generated by classical FGM results in a dangerous scenario where the obstacle and robot pass close to each other. Selecting the gaps considering their current and future sizes along with the right prediction will enable the FGM to select a safer gap, shown as dashed in Fig. 2, which avoids possible unsafe trajectories.

The classic FGM only calculates the final heading direction using the sensory information and it recursively uses these data to calculate the current gaps surrounding the robot. The approach is independent of the robot and the obstacles dynamics, so there is no calculation upon the velocity of mentioned parts of the environment. For this reason, the classic FGM sometimes faces unsafe trajectories as shown in Fig. 2. In the purposed approach, the robot uses the same sensory information to select the gaps considering the current and future size of the gaps and calculating the size change in the gaps.

The new FDGM gap selection approach is a two-stage procedure, shown in Fig. 3. Calculation of the possible prediction time based on time to collision is explained in section III-A, and the prediction of gap changes by the desired time is discussed in part III-B.

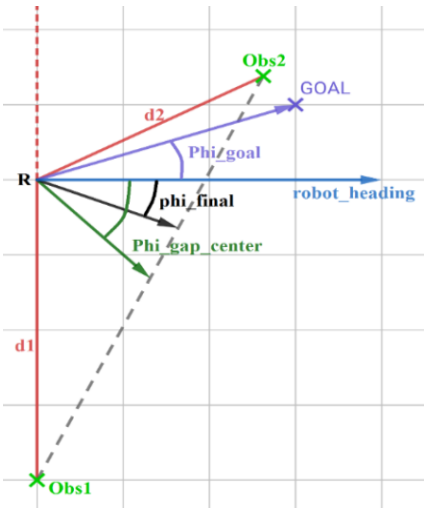


Figure 1. Robot-obstacle configuration, gaps, the midpoint of the largest gaps, goal point, and final heading angle.

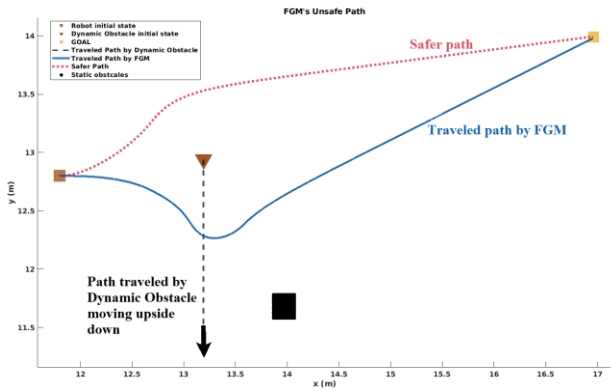


Figure 2. Robot-obstacle configuration with a dynamic obstacle moving with velocity vector V_1 in the free-space.

A. *Calculation of the Possible Prediction Time Based on Time to Collision*

The procedure of FDGM is based on the prediction of gap changes during the prediction time. This possible prediction time for each gap can be obtained taking into account the robot's velocity vector and how far the obstacles creating the gap borders are. Instead of selecting a fixed prediction time, the algorithm calculates the time at the point where the robot's heading line intersects the gap baseline. This calculated time which is shown as t_p in Fig. 4. is dependent on the velocity vector of the robot, and how far the robot is from the baseline of the gap. Its dynamic characteristics provide a better approach than the fixed prediction time approach.

Using a constant velocity model of two-dimensional motion, and considering the velocity vector of the robot ($\vec{V}_R$), the coordinate of the possible intersection point (P) where the robot's heading line crosses the gap baseline can be found using the Eqs. (2)-(3), where the robot's and the *i*th obstacle's current coordinates are considered as $\begin{bmatrix} x_R \\ y_R \end{bmatrix}$, and $\begin{bmatrix} x_{obsi} \\ y_{obsi} \end{bmatrix}$ respectively.

$$x_p = \frac{y_{obs1} - y_R + (\tan\theta \,.\, x_R - slope_{GB}.x_{obs1})}{\tan\theta - slope_{GB}} \quad (2)$$

$$y_p = slope_{GB}.x_p + y_{obs1} - slope_{GB}.x_{obs1} \quad (3)$$

($slope_{GB}$: The slope of the line between the Gap borders)

The Euclidian distance between the robot's current position and the point P can be obtained using the Eq. (4).

$$d_P = \sqrt{(x_p - x_R)^2 + (y_p - y_R)^2} \quad (4)$$

Finally, the prediction time (t_p) can be found as shown in Eq. (5).

$$t_p = \frac{d}{\vec{V}_R} \quad (5)$$

It's noteworthy to consider that in the cases where there is no intersection, the classic FGM would be used. An example of this configuration can be found in Fig. 6. For the rest of the procedure, Δt will be considered as t_p.

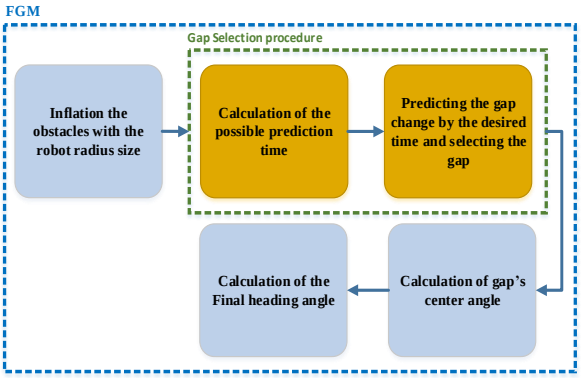


Figure 3. The new gap selection procedure

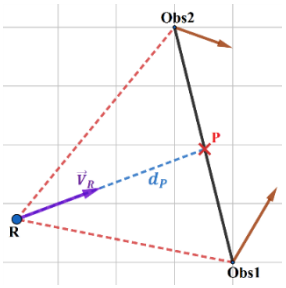


Figure 4. Robot-obstacle configuration for calculation of possible prediction time (R: robot, Obs1:obstacle, Obs2: obstacle 2, t_p: *prediction time*).

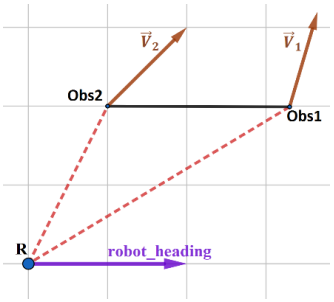


Figure 5. Example of the robot-obstacle configuration where there is no intersection.

B. *Predicting the Gap Change by the Desired Time*

The main step within the FDGM approach is predicting the gaps' future size using the calculated time from step 1.

Using the linear algebra concepts, the projected velocity vectors of the *i'th* gap border ($\vec{v}_{pi}$, i=1,2) onto the gap baseline, shown in Fig. 6, can be found using the Eq. (6).

$$\vec{v}_{pi} = \frac{\vec{b}.\vec{V}_i}{\|\vec{b}\|^2}\vec{b} \quad (6)$$

After obtaining the decomposed velocity vectors of the gap borders, the predicted size of the gap (φ_{PG}) represented in Fig. 8, can be calculated. The goal is to find the φ_{PG} in terms of measurable (x_{robot}, y_{robot}), d_1, d_2, θ_{11}, θ_{21}, V_1 and V_2 parameters coming from the robot's odometry and sensory data, shown in Fig. 8. d_1 and d_2 are distances to the gap borders, θ_{11}, θ_{21} are angles between the gap borders and the segment h, which

is the perpendicular line to the line between gap borders. Considering the fact that h crosses from the robot, it can be calculated using the law of Sines, as shown in Eq. (7).

$$\frac{sin\ (G_1)}{h} = \frac{sin\ (90^0)}{d_1} \rightarrow h = d_1.sin\ (G_1) \tag{7}$$

where: G1 is one of the angles of the triangle (R, GB1, GB2) represented in Fig. 7.

Considering the same triangle, the third side of the triangle (d_3) and the angles G_1 and G_2 can be calculated by using the law of Cosines, represented in Equations (8)-(10).

$$d_3 = \sqrt{d_1^2 + d_2^2 - 2d_1 d_2.cos(G_0)} \tag{8}$$

Where: G_0 is the current gap size

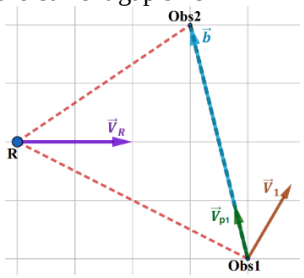


Figure 6. Vector decomposition using the projection of vectors for the first gap border (i=1) ($\vec{V}_1$: velocity vector of an obstacle creating the first gap border, $\vec{v}_{p1}$: the projected vector).

$$G_1 = cos^{-1}\left(\frac{d_1^2 + d_3^2 - d_2^2}{2.d_1.d_3}\right) \tag{9}$$

$$G_2 = cos^{-1}\left(\frac{d_2^2 + d_3^2 - d_1^2}{2.d_2.d_3}\right) \tag{10}$$

The angles between the gap borders and the perpendicular segment h can be calculated as follows:

$$\theta_{11} = 90^0 - G_1 \tag{11}$$

$$\theta_{21} = 90^0 - G_2 \tag{12}$$

By considering each side of the gap and writing the tangent formula for the sides, and using the constant velocity model of 2D motion for the robot-gap configuration, the displacement vectors of each gap from the point P (intersection of the perpendicular segment h and the segment pathing from the GB1 and GB2), as well as the displacement vectors from each side of the gap to their possible future position can be obtained as follows:

$$\tan\theta_{i1} = \frac{x_{i1}}{h} \rightarrow x_{i1} = h.\tan\theta_{i1} \tag{13}$$

$$\tan\theta_{i2} = \frac{x_{i2}}{h} \rightarrow x_{i2} = h.\tan\theta_{i2} \tag{14}$$

$$|\Delta x_i| = |x_{i2} - x_{i1}| \quad , i = 1,2 \tag{15}$$

Subtracting the Eq. (13) from Eq. (14), the following relationship can be obtained:

$$h.(\tan\theta_{i2} - \tan\theta_{i1}) = \Delta x_i \tag{16}$$

Considering the constant velocity model, Eq. (17), and replacing the Δx_i with the Eq. (16), the Eq. (18) can be obtained:

$$\Delta x_i = v_{pi}.\Delta t, \qquad \Delta t = t_p \tag{17}$$

$$h.(\tan\theta_{i2} - \tan\theta_{i1}) = v_{pi}.\Delta t \tag{18}$$

Using Eq. (18), the θ_{i2} can be found as follows:

$$\theta_{i2} = \tan^{-1}\left(\frac{v_{pi}\ .\ \Delta t}{h} + \tan\theta_{i1}\right) \tag{19}$$

Finally, using Eq. (19), the φ_{PG} can be calculated as follows:

$$\varphi_{PG} = \sum_{i=1}^{2} \theta_{i2} \tag{20}$$

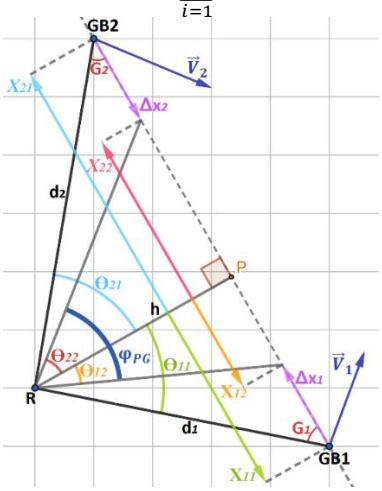


Figure 7. Robot-gap configuration for calculating the future gap changes.

The FDGM algorithm selects the largest predicted gap along with the other predicted gaps and passes it to the FGM. The simulation results of two different robot-obstacle configurations containing two dynamic obstacles and multiple static obstacles are illustrated in Fig 8. As shown in Fig. 8-a, the traveled paths for both FGM and FDGM algorithms from the starting time until time t_1 are the same.

At time t_1, the robot detects the second dynamic obstacle (DO2) which is moving downside up and the FGM selects the current largest available gap, but the FDGM calculates the gap's future size by considering prediction time and selects the largest predicted gap along with other predicted gaps, which consequently results in different trajectories for the mentioned algorithms in the rest of the travel. The result of this smart gap selection can be seen at the time t_2, where using the FGM causes the robot to be close to the DO2 while under the FDGM, the robot is in a safer path.

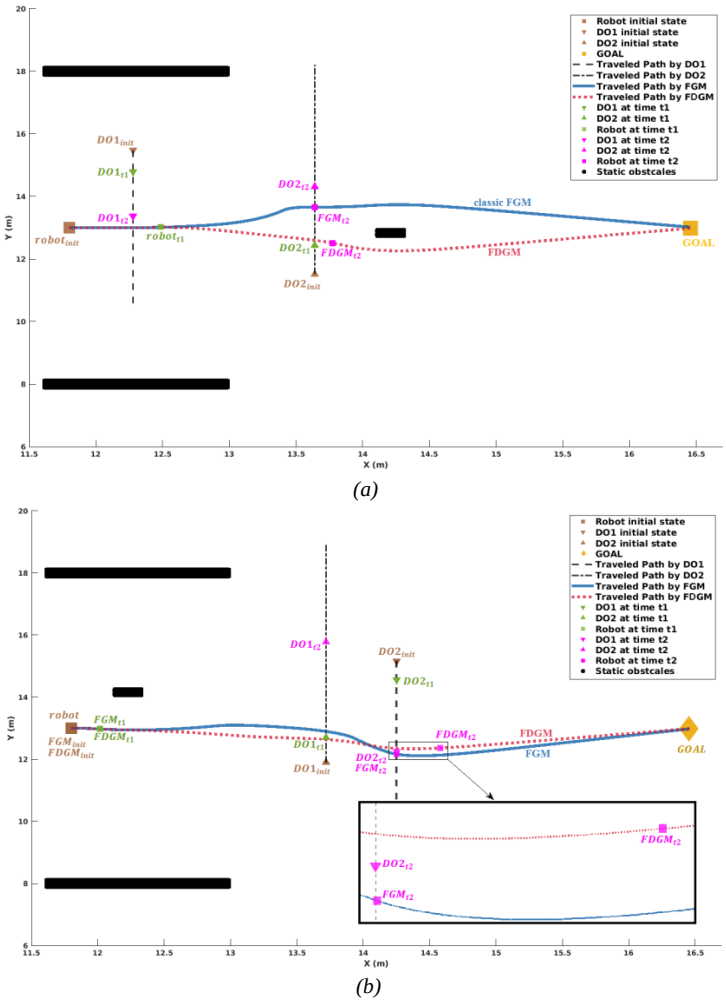
In Fig. 8-b, at time t_1, the FGM and FDGM select different gaps and the same consequences happen for the robot, where it gets too close to the DO2 with FGM, leading to an unsafe path.

IV. Simulation Results

In order to test the effectiveness of the proposed algorithm, a simulation environment is developed in Matlab® and the kinematic model of a differential drive robot equipped with a proportional heading angle controller is used to conduct the tests. The mentioned controller, shown in Eq. (21), drives the robot at constant linear velocity 0.15 m/s and the angular velocity, which is proportional to the angular difference between the robot's current heading angle, θ_{robot} and the FGM's final heading angle, φ_{final}.

$$V = Constant$$
$$w = K_p(\varphi_{final} - \theta_{robot}) \tag{21}$$

To assess the efficiency of the proposed algorithm, 300 Monte Carlo simulations are performed for FGM and FDGM, where the coordinate of obstacles are specified randomly. A LIDAR with a total 180˚ field of view (FOV) is used in the simulations and the total area of the environment is selected as 7m x 14m, and the initial and the goal coordinates are chosen as [11.8-13] and [16.5-13], respectively.



(a)

(b)

Figure 8. Comparison of the classic FGM and FDGM

The results are compared in terms of obstacle avoidance safety and length of the path. The metric used for the obstacle avoidance safety expressed in Eq. (22), is the same as used in [19], [20], and [24].

$$f(t) = \begin{cases} \dfrac{1}{d_{min}} - \dfrac{1}{d_0}, & for \quad d_{min} < d_0 \\ 0, & for \quad d_{min} \geq d_0 \end{cases} \tag{22}$$

where, d_{min} is the closest distance between the robot and the obstacles, and the scalar d_0 denotes the distance to an obstacle that imposes no danger for the robot during the travel. It is noteworthy that this scalar is set considering the environment and the robot dynamics. This safety metric is a function which is inversely proportional to the distance to the obstacle, d_{min}. So, comparing multiple scenarios, smaller values of safety-metric means it is safer.

The *p*th norm of any f(t) function can be calculated as:

$$\|f\|_p = (|f(t)|^p dt)^{1/p} \tag{23}$$

In this study, the infinity norm (p = ∞) of the obstacle avoidance safety metric is used to compare the most dangerous parts of each scenario for FGM and FDGM. Simulations are implemented to test the FGM and FDGM in the same environments using the 300 Monte Carlo

simulations in a computer with 16 GB of RAM, and an i7 4.6 GHz processor, using the Ubuntu 16.04 operating system. The average values of the obstacle avoidance safety metric and traveled distance values in the trajectories where the gap selection is different in FGM and FDGM are presented in Table I (α=40, d_0=2m).

TABLE I. MONTE CARLO SIMULATIONS RESULT

	Average safety metric	Average traveled distance (m)
FGM	0.0271	5.117
FDGM	0.0242	5.015

According to Table I, FDGM is 10.7% safer than the FGM, while the average travel distance values are almost the same. And this shows that the newly proposed method improves the safety aspect of the classic FGM without extending the total path.

V. CONCLUSION

This paper presents a novel approach to improve the gap selection procedure in classic FGM. The proposed model improves the gap selection quality of the classic FGM by selecting the gaps based on the prediction of the gap changes during the time. The proposed procedure is tested using the 300 Monto-Carlo simulations and the results demonstrate that the proposed model leads the robot to safer trajectories in comparison with the classic FGM. The next phase for this study includes implementing the FDGM on a physical platform and improving the prediction time by selecting more complex velocity models.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Hosein Houshyari and Volkan Sezer contributed to the analysis and implementation of the research. All the authors were involved in the drafting of the final manuscript. All authors had approved the final version.

ACKNOWLEDGMENT

This work was supported by the Turkish Scientific and Technological Research Council (TUBITAK) under project no 118E809.

REFERENCES

[1] A. Chakravarthy, D. Ghose, "Obstacle avoidance in a dynamic environment: a collision cone approach," *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, vol. 28, no. 5, pp. 562–574, 1998.

[2] K. Fujimura, *Motion Planning in Dynamic Environments*, Springer-Verlag, Tokyo, Japan, 1991.

[3] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968

[4] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, 1959.

[5] S. M. Lavalle, *Rapidly-exploring Random Trees: A New Tool for Path Planning*, Tech. Rep. (1998).

[6] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 846-894, 2011.

[7] R. Siegwart, I. R. Nourbakhsh, *Introduction to Autonomous Mobile Robots*, MIT Press, 2004.

[8] H. M. Choset, *Principles of Robot Motion: Theory, Algorithms, and Implementation*, MIT press, 2005.

[9] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *Autonomous Robot Vehicles*. Springer, 1986, pp. 396–404

[10] Y. Koren, J. Borenstein, "Potential field methods, and their inherent limitations for mobile robot navigation," *IEEE Conference on Robotics and Automation*, pp. 1398–1404, 1991.

[11] Q. Yao et al., "Path planning method with improved artificial potential field—a reinforcement learning perspective," *IEEE Access*, vol. 8, pp. 135513-135523, 2020.

[12] M. C. Lee and M. G. Park, "Artificial potential field based path planning for mobile robots using a virtual obstacle concept," in *Advanced Intelligent Mechatronics, 2003. AIM 2003. Proceedings. 2003 IEEE/ASME International Conference on*, vol. 2. IEEE, 2003, pp. 735–740

[13] J. Borenstein, Y. Koren, "Real-time obstacle avoidance for fast mobile robots," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 19, no. 5, pp. 1179–1187, 1989.

[14] J. Borenstein and Y. Koren, "The vector field histogram-fast obstacle avoidance for mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 278–288, 1991

[15] W. Chen, N. Wang, X. Liu, and C. Yang, "VFH* based local path planning for mobile robot," in *Proc. 2019 2nd China Symposium on Cognitive Computing and Hybrid Intelligence (CCHI)*, Xi'an, China, 2019, pp. 18-23.

[16] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics & Automation Magazine*, vol. 4, no. 1, pp. 23–33, 1997

[17] X. Zhong, J. Tian, H. Hu, et al. "Hybrid path planning based on safe A* algorithm and adaptive window approach for mobile robot in large-scale dynamic environment," *J Intell Robot Syst*, vol. 99, pp. 65–77, 2020.

[18] V. Sezer and M. Gokasan, "A novel obstacle avoidance algorithm: Follow the gap method," *Robotics and Autonomous Systems*, vol. 60, no. 9, pp. 1123–1134, 2012.

[19] M. Demir and V. Sezer, "Improved follow the gap method for obstacle avoidance," in *Proc. 2017 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, Munich, 2017, pp. 1435-1440.

[20] A. Özdemir and V. Sezer, "A hybrid obstacle avoidance method: follow the gap with dynamic window approach," in *Proc. 2017 First IEEE International Conference on Robotic Computing (IRC)*, Taichung, 2017, pp. 257-262.

[21] M. Demir and V. Sezer, (2019). Autonomous Overtaking Maneuver Design based on Follow the Gap Method. *ICINCO*.

[22] E. Çakmak, S. Tekin, A. Özdemir, and S. Boğosyan, "Gap based novel approach for safe and fast obstacle avoidance for autonomous platforms," in *Proc. 2020 IEEE 29th International Symposium on Industrial* Electronics (ISIE), Delft, Netherlands, 2020, pp. 1392-1397.

[23] M. Zohaib, S. M. Pasha, N. Javaid, A. Salaam, and J. Iqbal, "An improved algorithm for collision avoidance in environments having u and h shaped obstacles," *Studies in Informatics and Control*, vol. 23, no. 1, pp. 97–106, 2014.

[24] S. Nair and M. Kobilarov, "Collision avoidance norms in a trajectory planning," in *American Control Conference (ACC), 2011*. IEEE, 2011, pp. 4667–46

Hosein Houshyari is currently a M.Sc. student in Control & Automation Engineering Department at Istanbul Technical University, Istanbul, Turkey.
He is a researcher at Autonomous Mobility Group of ITU, working on TUBITAK's 118E809 project. His research areas include Autonomous robotics, Path planning, obstacle/collision avoidance, Localization, and SLAM.

Volkan Sezer received his BSc in Electronics and Telecommunication Engineering from Yildiz Technical University, in 2005, MSc in Mechatronics Engineering and PhD in Control and Automation Engineering from Istanbul Technical University, in 2008 and 2012 respectively. He is currently associate professor in Control & Automation Engineering Dep. at Istanbul Technical University, Turkey. His fields of interest are autonomous/semi-autonomous ground vehicles, mobile robots, simultaneous localization and mapping (SLAM), control of hybrid electric vehicles, trajectory planning, obstacle/collision avoidance, active safety and machine learning. He has authored or co-authored more than 60 research papers and managed several research projects on these subjects.