Crowd evacuation navigation for evasive maneuver of brownian based dynamic obstacles using reciprocal velocity obstacles

Susi Juniastuti[1], Moch Fachri[1], Fresy Nugroho[3], Supeno Mardi Susiki Nugroho[2], Mochamad Hariadi[2]
[1]Department of Electrical Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia
[2]Department of Computer Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia
[3]Department of Information Technology, Universitas Islam Negeri Maulana Malik Ibrahim, Malang, Indonesia

Article Info

Article history:

Received Mar 14, 2022
Revised May 15, 2022
Accepted Jun 8 ,2022

Keywords:

Brownian motion
Collision avoidance
Crowd evacuation
Mobile agents
Reciprocal velocity obstacles

ABSTRACT

This paper presents an approach for evasive maneuver against dynamic obstacles in multi-agent navigation in a crowd evacuation scenario. Our proposed approach is based on reciprocal velocity obstacles (RVO) with a different manner to treat the obstacles. We treat all possible hindrances in velocity space reciprocally thus all collision cones generated by other agents and obstacles are treated in the same RVO manner with the key difference in the effort of avoidance. Our approach assumes that dynamic obstacles bear no awareness of navigation space unlike agents thus the avoidance effort lies on behalf of the mobile agents, creating unmutual effort in an evasive maneuver. We display our approach in an evacuation scenario where a crowd of agents must navigate through an evacuation area trespassing zone filled with dynamic obstacles. These dynamic obstacles consist of random motion built based on Brownian motion thus posses an immense challenge for the mobile agent in order to overcome this hindrance and safely navigate to their evacuation area. Our experimentation shows that 51.1% fewer collisions occurred which is denote safer navigation for agents in approaching their evacuation point.

Corresponding Author:

Mochamad Hariadi
Department of Computer Engineering, Institut Teknologi Sepuluh Nopember
St. Teknik Kimia, Keputih, Surabaya, East Java 60111, Indonesia
Email: wichian.sit@mfu.ac.th

1. INTRODUCTION

Navigation of multi-agents in crowd evacuation holds an important key in agent-based model evacuation. This agent-based enables a high level of realism since it can model complex human behavior [1]. Multi-agent navigation under evacuation conditions is often required to provide necessary maneuvers to evade other agents and obstacles that exist on the evacuation route. This maneuver is handled by the agent's motion planning which relies on the agent's perception of its navigation space. In dealing with motion planning for agent navigation, many studies classify this type of planning into global and local planning [2]-[10]. Global planning depends heavily on the information of the current navigation plane that the agent currently navigating. This information is used for precalculation for path planning which is unsuitable for the dynamic environment in evacuation scenarios, where obstacles and agents' positions are constantly on move [3], [7], [10]-[19]. Global planning is only viable as long as the information of navigation space is available for the navigating agents. One example in a study about crowd simulation where evacuation path is set [20]. This type of planning is also applicable with information retrieved from surveillance [21] or shared through communication [22]. Such conditions cannot be fulfilled in a situation where evacuee find their position in some unknown environment [23]. Local planning takes information through agent local data while navigating

[24]–[31]. This makes the local planning method computationally efficient. The previous study we mention utilizes local planning for avoidance purposes against other dynamic objects. It is accomplished by using agent interact force based on the social force model. Whereas in uncertain environment issues, since information regarding the navigation space can merely be acquired locally, the function of local planning is even more critical [23]. An evacuation scenario is one such environment where obstacles are dynamically hurdled the evacuees, and other evacuees in such situations may hinder each other.

Velocity obstacles (VO) is a local motion planning invented for autonomous agents that facilitate safe agents' navigation. VO is able to predict the agents' velocity that will produce collisions using information obtainable from the sensors or other mean [32]. This basic method of VO has received much refinement through the years. Several instances to mention are non linear velocity obstacles (NLVO) [33], probabilistic velocity obstacles (PVO) [34], improved velocity obstacle (IVO) [35], finite-time velocity obstacle (FVO) [36], goal velocity obstacle (GVO) [37], inverse velocity obstacles (IVO) [38], probabilistic inverse velocity obstacle (PIVO) [39], and reciprocal velocity obstacles (RVO) [40]. NLVO built upon the contemplation of the nonlinear motion that may present in obstacles, it also introduces the notion of risk. The notion of risk allocates a value that depicts the risk characteristic to each velocity of the certain agent. The velocity that would produce instantaneous collision has more heightened values of risk than other velocities inside VO that do not instantly yield a collision in its course. PVO further expands the VO formula for uncertain assessment that may occur in sensors data acquisition. PVO consider that the mobile agent's data acquisition in real-life circumstances (e.g mobile robot) is taken from sensors with a particular boundary in their capability to thoroughly capture the essential perception of agents against its surrounding. This is since noise in sensors assessment may exist. This assembles a probabilistic framework to bridge the agent's perception and navigation for the avoidance purpose. IVO is devised with motion uncertainty of the obstacles and built an optimization objective function to enhance motion decisions. It's split the avoidance approach into two major operations, obstacles data processing for threat estimation and avoidance decision. The optimization objective function of IVO consists of the following parameters: risk of speed, the target speed deviation, and the collision time. FVO introduces time constraints [36]. It optimizes the operation of velocity alteration to minimize the avoidance velocity with the adequate velocity for agents to achieve their destination. Another extension called GVO brings the region of goal into VO formulation [37]. The goal region in GVO is the area of preferred velocities that will lead the agents toward their navigation objective. IVO introduces an ego-centric framework [38]. IVO presumes that each agent is the epicenter of the avoidance process at egoframe inside VO, thus assuming that the agent is stationary at the point of origin. At that assumption, proximate velocity beyond the VO collision cone is picked based on egocentric observation of the obstacle at two successive time instances. PIVO is based on IVO merged with PVO that notices uncertainty in assessment due to occurring noise. RVO is developed to deal reactive nature of avoidance in multiagent navigation and successfully prevent oscillatory motion that occurred during the reactive avoidance process [40].

The main problem with the existing velocity-based approach we mention lies in the capability of handling both reactive collision avoidance within agents and dynamic obstacles situations simultaneously. Some approaches were designed for avoidance against obstacles [33]-[37], while others optimized their effort against agents [38]-[40]. Using some method for handling obstacles avoidance in conjunction with other methods to handle agents avoidance, may break the collision-free properties of their respective methods. For the examples, the RVO is ensured to yield collision-free and oscillation-free properties as long as every agent makes equal avoidance reasoning [40]. In a crowd evacuation scenario, an agent is challenged not only by dynamic hindrances that exist in the environment but also by other agents that attempt to escape alongside. Thus, a method that can handle both cases with equally satisfactory performance is demanded.

In this paper, we presented our method based on RVO in an agent-based crowd evacuation. RVO is tasked to handle collision avoidance against agents and Brownian-based dynamic obstacles. Brownian-based dynamic obstacles are obstacles that behave erratically based on Brownian motion. Our approach is to treat the Brownian-based dynamic obstacles as agents that bear no awareness of its surrounding since they behave randomly as obstacles in an evacuation should be, a hindrance for the evacuee. Thus, we proposed different formulations in combined RVO used in multi-agent navigation. We proposed a combined reciprocal velocity obstacle for the agent as the union of the individual reciprocal velocity obstacles of the other agents including the obstacles but with a different value in the effort of avoidance, in contrast to the original RVO concept that differentiates VO generated from the agent and obstacles [40]. Mobile agents take the maximum effort of avoidance when dealing with Brownian-based dynamic obstacles due to the fact that obstacles cannot perform avoidance and serve only as hurdles. We conduct a crowd evacuation scenario filled with Brownian-based dynamic obstacles and hundreds of mobile agents to demonstrate the effectiveness of our proposed method.

2. METHOD

RVO is a concept based on velocity obstacles that take into account that the avoidance process between agents is a reactive process [40]. This method solved the velocity obstacle's major drawback in multi-agent navigation, which is oscillatory motion problems caused by reactive collision avoidance. Individually, the agent is expected to be autonomous in their navigation, all agents are anticipated to make the same logic in the collision avoidance method against each other. Avoidance velocity in RVO is performed by averaging the value of velocity that lies outside the other agent's velocity obstacle with the agent's existing velocity. In (1), shows the RVO of agent B to agent A, which is retains every velocity of agent A inside the velocity obstacles in $\mathrm{VO}_{\mathrm{B}}^{\mathrm{A}}(v_B)$averaged with the present velocity $\mathbf{v}_{\mathrm{A}}$ [40]. Which the $\mathrm{VO}_{\mathrm{B}}^{\mathrm{A}}(v_B)$ itself is velocity obstacle of agents B to agent A. It includes every probable velocity that will drive agent A to collide with B at a particular moment in the future.

$$\mathrm{RVO}_{\mathrm{B}}^{\mathrm{A}}(v_B, v_A) = \{v'_A | 2v'_A - v_A \in \mathrm{VO}_{\mathrm{B}}^{\mathrm{A}}(v_B)\} \quad (1)$$

Where; $\mathrm{RVO}_{\mathrm{B}}^{\mathrm{A}}$=reciprocal velocity obstacles of agent B to agent A
$\mathrm{VO}_{\mathrm{B}}^{\mathrm{A}}$=velocity obstacles of agent B to agent A
v_{A}=current velocity of agent A
v_{B}=current velocity of agent B

Our method involved the assumption that the Brownian-based dynamic obstacles have no awareness at all, thus the avoidance effort lies on behalf of the mobile agents. We treat all obstacles as absolute priority agents while the mobile agents' priority shared the same level of assigned value below the obstacles. The combined collision cones for an agent is equal to the union of RVO generated from avoidance against dynamic obstacles and other mobile agents. Our method distinction from the original RVO concept is in the combined reciprocal velocity obstacles whereas the original concept differentiates VO generated from the agent and obstacles [40]. In (2) shows the combined $\mathrm{RVO}^{\mathrm{i}}$ for agent $\mathrm{A_i}$ in out method. The velocities inside the collision cone are velocities that can produce collision in the future if the agents take those velocities. Every mobile agent's preferred velocities are set to the value that will guide them toward an evacuation point. The preferred speed in the direction of the target location is set uniformly at 3.7 unit per second based on the observed human speed [41]. In every cycle of motion planning, each agent's velocities may change due to evasive maneuvers against obstacles and other mobile agents. Ideally, the best velocity to take is $\boldsymbol{v}_i^{pref}$ as long as the preferred velocities do not produce collisions. If the v_i^{pref} is inside the collision cone in which is predicted to produce collision, the agent takes the closest valid velocity that is outside of the combined RVO for that agent.

$$RVO^i = \bigcup_{j \neq i} RVO_j^i(v_j, v_i, \alpha_j^i) \quad (2)$$

Where; RVO^i=combined reciprocal velocity obstacles agent Ai
RVO_j^i=reciprocal velocity obstacles of other agent or obstacles to agent Ai
v_i=current velocity of agent Ai
v_j =current velocity of other agent or obstacles
α_j^i=effort of avoidance of agent Ai in avoiding others

The environment which the mobile agents navigate may become dense with other mobile agents and dynamic obstacles. This create such a situation where the entire velocities of an agent can take become impossible to be admissible as it's predicted to cause a collision in the future. Based on the expected time to collision, we allow agents to take velocities inside the $\mathrm{RVO}^{\mathrm{i}}$ with a certain penalty. The velocity chosen if such condition happens is the velocity with minimal penalty among the velocities in $\mathrm{A}\mathbf{v}_{\mathrm{i}}$, thus the new velocities $\mathbf{v}'_{\mathrm{i}}$ for agent $\mathrm{A_i}$ is the velocity with lowest probable collision. The penalty value is given with (3). As the expected time to collision goes higher, the penalty value goes lower thus the velocities with the longest time to collision that is foreseen from certain dynamic obstacles or other mobile agents will be chosen. This creates more urgency to avoid the further immediate collision that will befall the agent. The lowest possible penalty value is at zero which is attained if the expected time to collision reaches infinite in which happens when no collision will transpire as there is no hindrance that causes it. The velocities, as well as the penalty value, are taken into account that the presence of the agents and dynamic obstacles in the neighboring region. This takes account due to the fact that the presence of other mobile agents and dynamic obstacles that are considerably distant from the agent's current position is not enough to contribute as a hindrance to the agent's navigation. We define the neighboring region around the current position of agents $\mathrm{A_i}$ with the radius of the region three times the agent's A_i radius.

Our Brownian based obstacles is build upon the fractal Brownian motion function [42], [43]. The mobile agents used in our experiment are presented in a circular shape as it is the optimal shape for mobile agents [40]. However, as shown in Figure 1, our dynamic obstacles vary in shape thus we opt for another method for shape approximation against dynamic obstacles. We use medial axis transform approximation [44] to approximate the shape of the dynamic obstacles. As mentioned before, the mobile agents take a complete effort in the avoidance process when dealing with obstacles. Figure 2 shows the various avoidance by mobile agents. Figure 2(a) is avoidance taken between mobile agents against an obstacle. Avoidance between mobile agents will result in mutual avoidance as shown with the line trace shown on Figure 2(b). When facing against the obstacle, the red line trace produced by a moving obstacle that translates in the opposite direction of the mobile agent shows no course alteration as shown in Figure 2(a), while the agent line trace shown by blue trace shows the evasive maneuver it takes to avoid the obstacles as shown in Figure 2(b).

$$penalty(\boldsymbol{v}'_i) = \frac{1}{\mathrm{tc}_i(\boldsymbol{v}'_i)} + \left\|\boldsymbol{v}_i^{pref} - \boldsymbol{v}'_i\right\| \tag{3}$$

Where; $\mathrm{penalty}(\mathrm{v}'_\mathrm{i})$=penalty value for velocity v'_i
$\mathrm{tc}_\mathrm{i}(\mathrm{v}'_\mathrm{i})$ =expected time to collision for velocity v'_i
$\mathrm{v}_\mathrm{i}^\mathrm{pref}$=preference velocity of agent Ai
v'_i=velocity candidate to be chosen for agent Ai avoidance

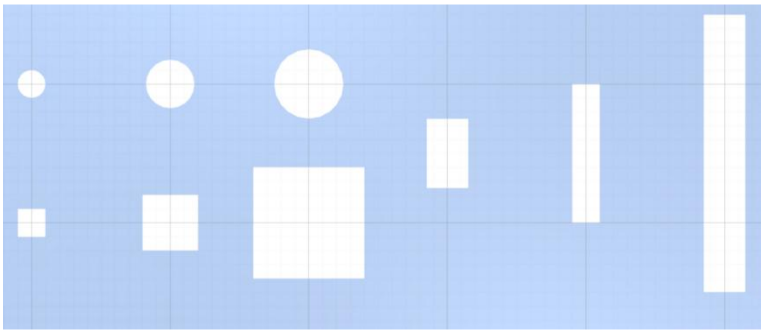
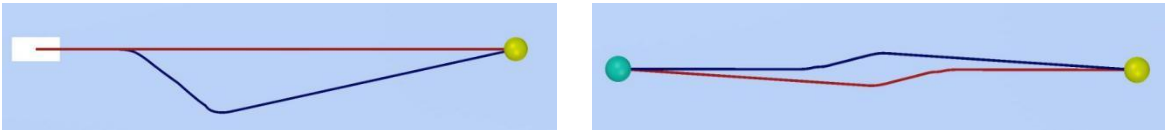


Figure 1. Dynamic obstacles with various shape and size



(a) (b)

Figure 2. Avoidance between various agents (a) avoidance between mobile agent against an obstacle and (b) avoidance between mobile agents

3. RESULTS AND DISCUSSION

We tested our proposed method with an evacuation scenario in the environment shown in Figure 3. All mobile agents start at the area labeled A in Figure 3 and are tasked to navigate toward the evacuation area shown with label B. In the middle of the navigation plane, we put a massive amount of Brownian-based dynamic obstacles with varying sizes, shapes, and parameters. Figure 1 shows the dynamic obstacles that appeared in our experiment with shapes that vary to circle, cube, and long cube shape, also three different sizes. We conducted this scenario from total agents from 100 to 1000 with our proposed method and the original RVO method as a comparison. In Figure 4, we demonstrate 500 mobile agents with our implementation of our proposed method. The agents moving through scattered dynamic obstacles. The result shown in Figure 5 and detailed result in Table 1 give us information of collision occurred in our experimentation. The majority of collision is caused by agents' collision toward obstacles as it is the most challenging maneuver due to the fact that the Brownian-based dynamic obstacles behave unpredictably unlike agents to agent avoidance.

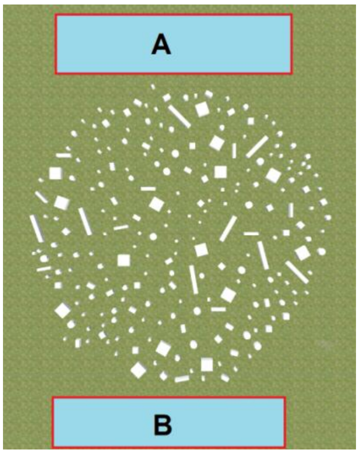


Figure 3. Navigation space environment

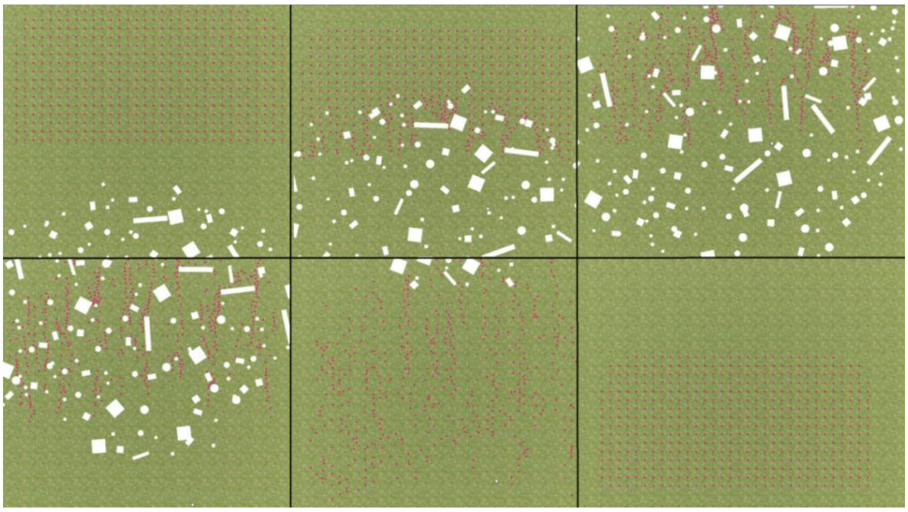
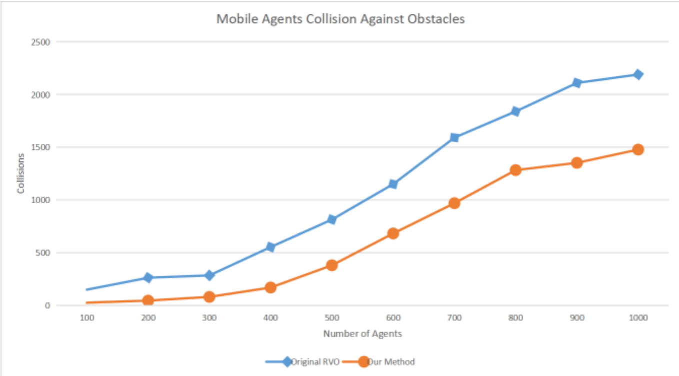


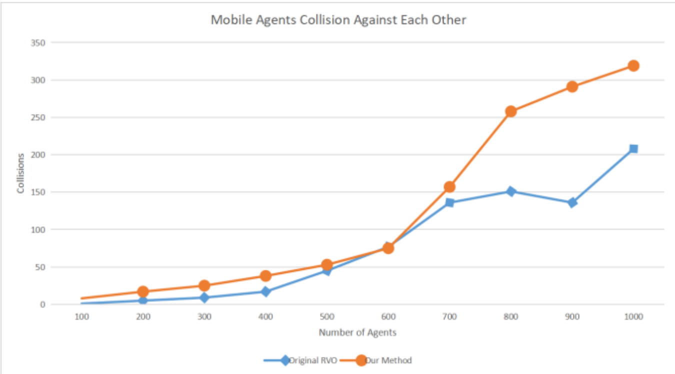
Figure 4. 500 mobile agents moving through scattered brownian based dynamic obtacles

Our proposed method is shown to produce better maneuver, it's caused 46.09% less collision in average compared to the original RVO method as shown in graphics on Figure 5(a). However there are several drawbacks as shown in Figure 5(b). Our method caused mobile agents' collisions against each other to occur more often. Our approach to treating all obstacles as absolute priority agents caused the velocity obstacles produced by Brownian-based dynamic obstacles to have a higher penalty value. This is further reinforced by another agent's direction that shares the general direction toward the evacuation point of area B in our navigation space. This situation creates more reason for agents to bump at each other when evacuating. The obstacles may approach agents in the opposite direction of agents' evacuation point, therefore creating more immediate collision compared to other agents that navigate in the same direction. This collision event happens infrequently as opposed to collision against obstacles, which in turn generated the overall collision shown in Figure 5(c) that indicates that our proposed method produces a safer outcome.

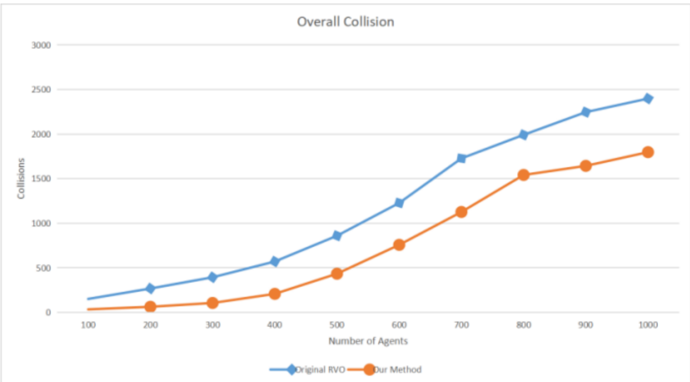
While our method can produce safer navigation as indicated by collision occurrence, the average time to reach destination shown in Figure 5(d) annunciates less efficient navigation. The time required for the agents to arrive at the evacuation point increased by an average of 1.33 seconds. We perceive this as an important matter for navigation in general, especially in an evacuation scenario. As stated by Godoy *et al.* [45], many studies forget about efficient navigation while achieving safe navigation. Our findings will be further investigated in future studies to improve efficiency while still paying attention to navigation safety.



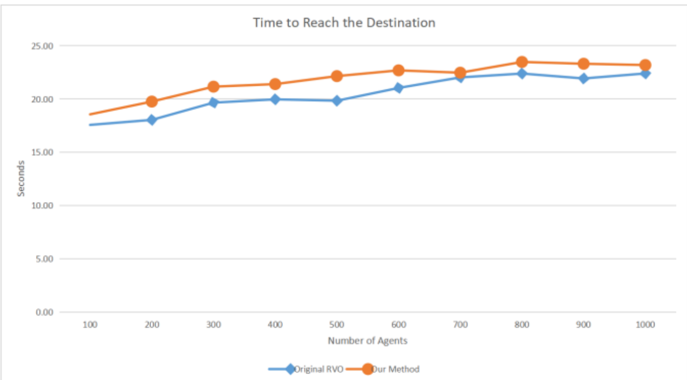
(a)



(b)



(c)



(d)

Figure 5. Result of experiment in graph (a) mobile agents collisions against obstacles (b) mobile agents collisions against each other (c) overall collisions of every agents, and (d) time to reach the destination

Table 1. Result of experiment

Number of agents	Brownian motion based dynamic obstacles as agents: Total of collisions	Mobile agents collisions against obstacles	Mobile agents collisions against each other	Average time to reach destination (seconds)	Generic Brownian motion based dynamic obstacles: Total of collisions	Mobile agents collisions against obstacles	Mobile agents collisions against each other	Average time to reach destination (seconds)
100	33	25	8	18.56	150	149	1	17.57
200	62	45	17	19.76	267	262	5	18.04
300	105	80	25	21.16	393	284	9	19.66
400	207	169	38	21.40	570	553	17	19.97
500	433	380	53	22.15	859	814	45	19.84
600	757	682	75	22.69	1227	1150	77	21.04
700	1126	969	157	22.47	1727	1591	136	22.03
800	1541	1283	258	23.48	1991	1840	151	22.39
900	1643	1352	291	23.31	2246	2110	136	21.93
1000	1797	1478	319	23.19	2398	2190	208	22.40

4. CONCLUSION

Our proposed method was able to improve agents' avoidance maneuver against obstacles that dynamically shift based on Brownian motion. Using the approach of treating all obstacles as absolute priority agents caused mobile agents'collision against obstacles to lessen on average by 46.09%. The overall collision that occurred in our experimentation show 51.1% less collision which is denote safer navigation for agents in approaching their evacuation point. However, the average time to reach the destination is increased by an average of 1.33 seconds. Our further study will focus on generating efficient navigation, which is not only secure in regard to the collision but also generates faster time to reach the evacuation point.

REFERENCES

[1] V. Kountouriotis, S. C. Thomopoulos, and Y. Papelis, "An agent-based crowd behaviour model for real time crowd behaviour simulation," *Pattern Recognition Letters*, vol. 44, pp. 30–38, Jul. 2014. doi: 10.1016/j.patrec.2013.10.024.

[2] Y. Hong, S. Kim, and J. Cha, "Integrated Global and Local Path Planning for Quadrotor Using Particle Swarm Optimization," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 15621–15625, 2020. doi: 10.1016/j.ifacol.2020.12.2497.

[3] J. Barraquand and J.-C. Latombe, "Robot motion planning: A distributed representation approach," *International Journal of Robotic Research - IJRR*, vol. 10, pp. 628–649, 12 1991. doi: 10.1177/027836499101000604.

[4] B. Shi, Y. Su, C. Wang, L. Wan, and Y. Luo, "Study on intelligent collision avoidance and recovery pathplanning system for the waterjet-propelled unmanned surface vehicle," *Ocean Engineering*, vol. 182, pp. 489–498, Jun. 2019. doi: 10.1016/j.oceaneng.2019.04.076.

[5] A. Ahmed, A. Maged, A. Soliman, H. El-Hussieny, and M. Magdy, "Space deformation based path planning for Mobile Robots," *ISA Transactions*, p. S001905782100447X, Aug. 2021. doi: 10.1016/j.isatra.2021.08.019.

[6] N. Wang and H. Xu, "Dynamics-Constrained Global-Local Hybrid Path Planning of an Autonomous Surface Vehicle," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 7, pp. 6928–6942, Jul. 2020. doi: 10.1109/TVT.2020.2991220.

[7] C. Cheng, Q. Sha, B. He, and G. Li, "Path planning and obstacle avoidance for AUV: A review," *Ocean Engineering*, vol. 235, p. 109355, Sep. 2021. doi: 10.1016/j.oceaneng.2021.109355.

[8] Y. Tang, Y. Miao, A. Barnawi, B. Alzahrani, R. Alotaibi, and K. Hwang, "A joint global and local path planning optimization for UAV task scheduling towards crowd air monitoring," *Computer Networks*, vol. 193, p.107913, Jul. 2021. doi: 10.1016/j.comnet.2021.107913.

[9] D. Lyu, Z. Chen, Z. Cai, and S. Piao, "Robot path planning by leveraging the graph-encoded Floyd algorithm," *Future Generation Computer Systems*, vol. 122, pp. 204–208, Sep. 2021. doi: 10.1016/j.future.2021.03.007.

[10] Z. Chen, Y. Zhang, Y. Zhang, Y. Nie, J. Tang, and S. Zhu, "A Hybrid Path Planning Algorithm for Unmanned Surface Vehicles in Complex Environment with Dynamic Obstacles," *IEEE Access*, vol. 7, pp. 126439–126449, 2019. doi: 10.1109/ACCESS.2019.2936689.

[11] T. T. Mac, C. Copot, D. T. Tran, and R. D. Keyser, "A hierarchical global path planning approach for mobile robots based on multi-objective particle swarm optimization," *Applied Soft Computing*, vol. 59, pp. 68–76, Oct. 2017. doi: 10.1016/j.asoc.2017.05.012.

[12] K. Yu *et al.*, "USV path planning method with velocity variation and global optimisation based on AIS service platform," *Ocean Engineering*, vol. 236, p. 109560, Sep. 2021. doi: 10.1016/j.oceaneng.2021.109560.

[13] X. Guo, M. Ji, Z. Zhao, D. Wen, and W. Zhang, "Global path planning and multi-objective path control for unmanned surface vehicle based on modified particle swarm optimization (PSO) algorithm," *Ocean Engineering*, vol. 216, p. 107693, Nov. 2020. doi: 10.1016/j.oceaneng.2020.107693.

[14] P. Chen, Y. Huang, E. Papadimitriou, J. Mou, and P. van Gelder, "Global path planning for autonomous ship: A hybrid approach of Fast Marching Square and velocity obstacles methods," *Ocean Engineering*, vol. 214, p. 107793, Oct.2020. doi: 10.1016/j.oceaneng.2020.107793.

[15] S. Han, L. Wang, Y. Wang, and H. He, "An efficient motion planning based on grid map: Predicted Trajectory Approach with global path guiding," *Ocean Engineering*, vol. 238, p. 109696, Oct. 2021. doi: 10.1016/j.oceaneng.2021.109696.

[16] C. D. B. Borges, A. M. A. Almeida, I. C. P. Júnior, and J. J. d. M. Sá Junior, "A strategy and evaluation method for ground global path planning based on aerial images," *Expert Systems with Applications*, vol. 137, pp. 232–252, Dec. 2019. doi: 10.1016/j.eswa.2019.06.067.

[17] P. Chand and D. A. Carnegie, "A two-tiered global path planning strategy for limited memory mobile robots," *Robotics and Autonomous Systems*, vol. 60, no. 2, pp. 309–321, Feb. 2012. doi: 10.1016/j.robot.2011.11.009.

[18] S. Aggarwal and N. Kumar, "Path planning techniques for unmanned aerial vehicles: A review, solutions, and challenges," *Computer Communications*, vol. 149, pp. 270–299, Jan. 2020. doi: 10.1016/j.comcom.2019.10.014.

[19] G. Xie, X. Du, S. Li, J. Yang, X. Hei, and T. Wen, "An efficient and global interactive optimization methodology for path planning with multiple routing constraints," *ISA Transactions*, p. S0019057821001907, Mar. 2021, doi: 10.1016/j.isatra.2021.03.041.

[20] M. N. bin Jamaludin *et al.*, "OpenGL 3D crowd evacuation simulation at universiti tun hussein onn malaysia (UTHM) hostel," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 16, no. 2, pp. 1034-1041, Nov. 2019. doi: 10.11591/ijeecs.v16.i2.pp1034-1041.

[21] M. R. Bhuiyan *et al.*, "Hajj pilgrimage video analytics using CNN," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 5, pp. 2598–2606, Oct. 2021, doi: 10.11591/eei.v10i5.2361.

[22] P. N. F. M. Shamsuddin, R. M. Ramli, and M. A. Mansor, "Navigation and motion control techniques for surface unmanned vehicle and autonomous ground vehicle: a review," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 4, pp. 1893–1904, Aug. 2021, doi: 10.11591/eei.v10i4.3086.

[23] S. H. Abdulredah and D. J. Kadhim, "Developing a real time navigation for the mobile robots at unknown environments," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 20, no. 1, p. 500, Oct. 2020. doi: 10.11591/ijeecs.v20.i1.pp500-509.

[24] H. Simas and R. D. Gregorio, "Smooth transition for collision avoidance of redundant robots: An on-line polynomial approach," *Robotics and Computer-Integrated Manufacturing*, vol. 72, p. 102087, Dec. 2021. doi: 10.1016/j.rcim.2020.102087.

[25] B. B. K. Ayawli, A. Y. Appiah, I. K. Nti, F. Kyeremeh, and E. I. Ayawli, "Path planning for mobile robots using Morphological Dilation Voronoi Diagram Roadmap algorithm," *Scientific African*, vol. 12, p. e00745, Jul. 2021. doi: 10.1016/j.sciaf.2021.e00745.

[26] H. Sang, Y. You, X. Sun, Y. Zhou, and F. Liu, "The hybrid path planning algorithm based on improved A* and artificial potential field for unmanned surface vehicle formations," *Ocean Engineering*, vol. 223, p. 108709, Mar. 2021. doi: 10.1016/j.oceaneng.2021.108709.

[27] X. Zhang, C. Wang, L. Jiang, L. An, and R. Yang, "Collision-avoidance navigation systems for Maritime Autonomous Surface Ships: A state of the art survey," *Ocean Engineering*, vol. 235, p. 109380, Sep. 2021. doi: 10.1016/j.oceaneng.2021.109380.

[28] Z. Wang, G. Li, and J. Ren, "Dynamic path planning for unmanned surface vehicle in complex offshore areas based on hybrid algorithm," *Computer Communications*, vol. 166, pp. 49–56, Jan. 2021. doi: 10.1016/j.comcom.2020.11.012.

[29] H. Niu, C. Ma, and P. Han, "Directional optimal reciprocal collision avoidance," *Robotics and Autonomous Systems*, vol. 136, p. 103705, Feb. 2021. doi: 10.1016/j.robot.2020.103705.

[30] L. Li, D. Wu, Y. Huang, and Z.-M. Yuan, "A path planning strategy unified with a COLREGS collision avoidance function based on deep reinforcement learning and artificial potential field," *Applied Ocean Research*, vol. 113, p. 102759, Aug. 2021. doi: 10.1016/j.apor.2021.102759.

[31] Y. Shin and E. Kim, "Hybrid path planning using positioning risk and artificial potential fields," *Aerospace Science and Technology*, vol. 112, p.106640, May 2021. doi: 10.1016/j.ast.2021.106640.

[32] P. Fiorini and Z. Shiller, "Motion Planning in Dynamic Environments Using Velocity Obstacles," *The International Journal of Robotics Research*, vol. 17, no. 7, pp. 760–772, Jul. 1998. doi: 10.1177/027836499801700706.

[33] F. Large, S. Sckhavat, Z. Shiller, and C. Laugier, "Using non-linear velocity obstacles to plan motions in a dynamic environment," in *7th International Conference on Control, Automation, Robotics and Vision, 2002. ICARCV 2002*, pp. 734-739 vol. 2, doi: 10.1109/ICARCV.2002.1238513.

[34] C. Fulgenzi, A. Spalanzani, and C. Laugier, "Dynamic Obstacle Avoidance in uncertain environment combining PVOs and Occupancy Grid," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, 2007, pp. 1610-1616, doi: 10.1109/ROBOT.2007.363554.

[35] W. Zhang, S. Wei, Y. Teng, J. Zhang, X. Wang, and Z. Yan, "Dynamic Obstacle Avoidance for Unmanned Underwater Vehicles Based on an Improved Velocity Obstacle Method," *Sensors*, vol. 17, no. 12, p. 2742, Nov. 2017. doi: 10.3390/s17122742.

[36] S. Samavati, M. Zarei, and M. T. Masouleh, "An optimal motion planning and obstacle avoidance algorithm based on the finite time velocity obstacle approach," in *2017 Artificial Intelligence and Signal Processing Conference (AISP)*, 2017, pp. 250-255, doi: 10.1109/AISP.2017.8324091.

[37] J. Snape and D. Manocha, "Goal velocity obstacles for spatial navigation of multiple virtual agents," in *12th International Conference on Autonomous Agents and Multiagent Systems 2013, AAMAS 2013*, vol. 2, pp.1191–1192, 05 2013. doi:10.5555/2484920.2485137.

[38] P. S. N. Jyotish, Y. Goel, A. V. S. S. B. Kumar, and K. M. Krishna, "IVO: Inverse Velocity Obstacles for Real Time Navigation," in *AIR 2019: Proceedings of the Advances in Robotics 2019*, July 2019. doi: 10.1145/3352593.3352610.

[39] P. S. Naga Jyotish, Y. Goel, A. V. S. Sai Bhargav Kumar and K. M. Krishna, "PIVO: Probabilistic Inverse Velocity Obstacle for Navigation under Uncertainty," in *2019 28th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, 2019, pp. 1-6, doi: 10.1109/RO-MAN46459.2019.8956406.

[40] J. van den Berg, Ming Lin, and D. Manocha, "Reciprocal Velocity Obstacles for real-time multi-agent navigation," in *2008 IEEE International Conference on Robotics and Automation*, 2008, pp. 1928-1935, doi: 10.1109/ROBOT.2008.4543489.

[41] K. L. Steudel-Numbers and C. M. Wall-Scheffler, "Optimal running speed and the evolution of hominin hunting strategies," *Journal of Human Evolution*, vol. 56, no. 4, pp. 355–360, Apr. 2009. doi: 10.1016/j.jhevol.2008.11.002.

[42] V. Mitic *et al.*, "Brownian motion and fractal nature," *Modern Physics Letters B*, vol. 34, no. 19n20, p. 2040061, Jul. 2020. doi: 10.1142/S0217984920400618.

[43] W. Song, M. Li, Y. Li, C. Cattani, and C.- H. Chi, "Fractional Brownian motion: Difference iterative forecasting models," *Chaos, Solitons & Fractals*, vol. 123, pp. 347–355, Jun. 2019, doi: 10.1016/j.chaos.2019.04.021.

[44] Y. Ma, D. Manocha, and W. Wang, "Efficient Reciprocal Collision Avoidance between Heterogeneous Agents Using CTMAT," in *Proceedings of the 17th International Conference on Autonomous Agents and Multi Agent Systems*, July. 2018. doi: 10.5555/3237383.3237853.

[45] J. Godoy, S. J. Guy, M. Gini, and I. Karamouzas, "C-Nav: Distributed coordination in crowded multi-agent navigation," *Robotics and Autonomous Systems*, vol. 133, p. 103631, Nov. 2020. doi: 10.1016/j.robot.2020.103631.

BIOGRAPHIES OF AUTHORS

Susi Juniastuti is received a Master's degree in Advanced Applied Science from Tokyo Institute of Technology, Japan in 2005. She obtained Bachelor's Degree in Electronics Engineering from Institut Teknologi Sepuluh Nopember, Indonesia in 1994. She is currently a doctoral student of the Electrical Engineering Department of Institut Teknologi Sepuluh Nopember, Indonesia. Her research interest is in the fields of digital systems, signal processing, and intelligent system. She is a student member of IEEE. She can be contacted at email: susi@te.its.ac.id.

Moch Fachri received both Bachelor's and Master's degrees in Electrical Engineering Department of Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia, in 2016 and 2018 respectively. He is currently a doctoral student of the Electrical Engineering Department of Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia. He is also a student member of IEEE. His research interest is in game technology, mobile agents navigation, crowd modeling, and intelligent system. He can be contacted at email: 7022211027@mhs.its.ac.id.

Fresy Nugroho received a Bachelor's degree in the Dept of Electrical Engineering of Univ. Brawijaya, Malang, Indonesia, in 1997. He is currently working towards a Ph.D. degree with the Dept of Electrical Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia. His research interests include game technology, education, computer vision, optimization. He can be contacted at email: fresy@ti.uin-malang.ac.id.

Supeno Mardi Susiki Nugroho received the bachelor's degree in Electrical Engineering from the Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia, the master's degree from the Department of Informatics Engineering, Institut Teknologi Bandung, and the Ph.D. degree in electrical engineering from the Institut Teknologi Sepuluh Nopember. He is currently the head of the computer engineering department, Institut Teknologi Sepuluh Nopember. His research interests include image processing, serious game, data mining, and intelligent systems. He is a member of IEEE. He can be contacted at email: mardi@its.ac.id.

Mochamad Hariadi received the B.E. degree in Electrical Engineering Department of Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia, in 1995. He received both M.Sc. and Ph.D. degrees in Graduate School of Information Science Tohoku University Japan, in 2003 and 2006 respectively. Currently, he is the staff of Electrical Engineering Deparment of Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia. His research interest is in video and image processing, data mining, big data, and intelligent system. He is the project leader in joint research with PREDICT JICA project Japan and WINDS project Japan. He is a member of IEEE and a member of IEICE. He can be contacted at email: mochar@te.its.ac.id.