Fusion Bacterial Foraging and Particle Swarm Optimization Algorithm For An Autonomous Hovercraft Control

Journal:	*Part C: Journal of Mechanical Engineering Science*
Manuscript ID	JMES-22-2936.R2
Manuscript Type:	Original Research Article
Date Submitted by the Author:	28-May-2023
Complete List of Authors:	Tran, Khoa; Tamkang University College of Business and Management Pham, Duc Lam; Faculty of Engineering and Technology, Nguyen Tat Thanh University Tran, Thanh Trang; Faculty of Electrical and Electronics Engineering, Ho Chi Minh City University of Food Industry Nguyen, Hoang Nam; Modeling Evolutionary Algorithms Simulation and Artificial Intelligence, Faculty of Electrical and Electronics Engineering, Ton Duc Thang University
Keywords:	Control Engineering, Automobile, Motion Control, Vehicles, Particle Swarm Optimization
Abstract:	This article proposes a novel method for designing an effective autonomous Hovercraft vehicle controller. In the first step, we design a PD like Fuzzy controller, in which Proportional and Derivative (PD) parameters are used as the Fuzzy Scaling Factor of the controller design. Next, Bacterial Foraging (BF) and Particle Swarm Optimization (PSO) are combined to create a fusion algorithm to optimize the proportional and derivative gains of the PD like Fuzzy controller. The integral of time multiplied square error (ITSE) fitness function is applied to find the minima of the controller design error criterion. Numerical simulations are then implemented on a real autonomous Hovercraft motion model. Results indicate that the proposed controller design is more stable, reliable and accurate than existing controllers.

Fusion Bacterial Foraging and Particle Swarm Optimization Algorithm For An Autonomous Hovercraft Control

Huu Khoa Tran[1], Pham Duc Lam[2], Tran Thanh Trang[3], and Hoang Nam Nguyen[4,*]

[1] Department of Information Management, Tamkang University, New Taipei City, Taiwan; email: hk5tran@mail.tku.edu.tw, https://orcid.org/0000-0001-5268-2355
[2] Faculty of Engineering and Technology, Nguyen Tat Thanh University, Vietnam email: pdlam@nttu.edu.vn
[3] Faculty of Electrical and Electronics Engineering, Ho Chi Minh City University of Food Industry, Ho Chi Minh City, Vietnam; trangtt@hufi.edu.vn
[4*] Modeling Evolutionary Algorithms Simulation and Artificial Intelligence, Faculty of Electrical and Electronics Engineering, Ton Duc Thang University, 19 Nguyen Huu Tho Street, Tan Phong Ward, District 7, Ho Chi Minh City, Vietnam;
* Correspondence: nguyenhoangnam@tdtu.edu.vn.

Abstract: In this article, we proposed a novel method for designing a controller for an autonomous hovercraft vehicle. In the first step, we design a PD like Fuzzy control, in which Proportional and Derivative (PD) parameters are used as the Fuzzy Scaling Factor (FSF) of the controller design. Next, the Particle Swarm Optimization (PSO) improved by Fusion Bacterial Foraging (BF) (named hereafter FBF-PSO) is proposed to optimize the proportional and derivative gains of the PD like Fuzzy controller. The integral of time multiplied square error (ITSE) fitness function was applied to find the minima of controller design criterion. Numerical simulations are then implemented on a real autonomous hovercraft motion model. Results show that the new controller design is more stable, reliable and accurate than available controllers of the real hovercraft mathematical model.

Keywords: Bacterial Foraging (BF) Algorithm, Particle Swarm Optimization (PSO), PD like Fuzzy, AGV Hovercraft, Integral of Time Multiplied Square Error (ITSE).

1. Introduction

In recent years, many researchers have focused on using nonlinear control to build an autonomous hovercraft [1–6] that can work well on various surfaces such as water, sand beach, ice and especially on muddy land. Most of these available controllers of the real hovercraft models are just based on the traditional PD controller and very few methods of optimizing the controller parameters are introduced for this system. The main aim of this paper is to propose a new way to optimize the proportional and derivative gains of the PD controller of the hovercraft.

Based on expert knowledge system, the Fuzzy Logic Controllers (FLC) have the advantage solution to the issues that could realize by human operators. Many works [7-15] has utilize this technique to solve various control system subjects and obtain considerably better results compare to others. When physical process has uncertain information and limited data, Fuzzy rule-based model is an appropriate choice. Therefore, we also use this approach in the hovercraft model control system. In this paper, we design a set of Fuzzy triangular membership functions and optimizes Scaling Factors (SF) as the inputs of Fuzzy controller. Although evolutionary algorithms [16, 17] such as Particle Swarm Optimization (PSO), Simulated Annealing (SA) and Genetic Algorithm (GA), can easily detect the local minima but it is difficult to detect the global optimization. Particle swarm optimization (PSO) [18-22] is a population-based heuristic algorithm that is invented based on observing animal swarm social behavior to detect accurate targets in multi-dimensional space. PSO uses particles (individuals), which are iteratively updated in each iteration, to perform searches. For finding the optimal solution, each particle decides its search direction based on its best previous location (cognitive part) and all other members' best locations (social part). In 2002, K. M. Passino [23], inspired from the Bacteria Foraging (BF) behavior of *Escherichia-coli* bacteria, firstly proposed a bionic algorithm that is very effective for distributed optimization and control design. Although, BF algorithm is very effective in local search but it is easily tapped or delayed in reaching global solution because of the random search directions during chemotaxis process. Due to eliminating the local optima trap and improving the optimal process, the modern engineering application researchers have introduced various PSO-based algorithms to solve the problems, for instance: GA and PSO [24], PSO and SA [25], hybrid PSO-Differential Evolution [26], direct search method (DSM) and hybrid PSO [27], hybrid Swam Intelligence approach [28]. The above techniques can expand the optimal convergence pace by taking the advantages while eliminating the weakness of the solo algorithms.

Thus, we proposed a fusion algorithm, which is called as fusion Bacterial Foraging and Particle Swarm Optimization (fBF-PSO) to adjust the global search efficiency and increase the convergence speed in shorter operated generation, in this article. We utilize PD control to enhance the hovercraft system stability. Additionally, Fuzzy Inference System (FIS) is utilized for reducing the steady state error. The fusion BF-PSO technique is also used to improve the performance of this PD like Fuzzy control by minimizing the ITSE fitness cost function [29-32]. The proposed controller can obtain the high-performance control of the Hovercraft in terms of higher accuracy, quick response, high stability and smoother maneuver. To the best of our knowledge, this is the first paper that combine the advantages of PSO and BF method in designing an effective and optimal PD controller for a real hovercraft model. In section 2, we describe the hovercraft configuration. In section 3, optimization algorithm and control strategy are introduced. In section 4, numerical simulation control of a real hovercraft model is presented. Finally, conclusions are discussed in section 5.

2. Autonomous Hovercraft configuration

The air cushion vehicle (ACV), which is usually called as hovercraft, is composed of rotors and a cushion. Blower (rotor duct fan) is used to provide a large volume of air inside the air cushion, which is larger than the atmospheric pressure. Therefore, the hovercraft can float and is capable of travelling over ice, water, land, mud… [1-6, 13]. A tilt servo motor, which is located at the rear, is usually employed to steer the hovercraft. Although many modern technologies are utilized, there is still needs for a more advanced hovercraft maneuvering system with better performance. Especially, faster response and higher reliability are of great interest. Figure 1 shows a popular hovercraft model used in this research, which has a single tilt servo motor, a propeller settled along z-axis and a blower attached along y-axis.

The dynamic Hovercraft model utilized in this paper is referenced from [1, 6 and 13] with right hand coordinate systems. X-axis is used to position the lateral direction for controlling the sway motion and surge position. The Z-axis is in vertical direction and its positive direction is downwards. Whilst Y-axis is along its body, which is necessary for controlling sway motion or surge position. The hovercraft's kinematics can be described as equation (1):

$$\begin{cases} \dot{x} = p\cos\varphi - s\sin\varphi \\ \dot{y} = q\cos\varphi - s\sin\varphi \\ \dot{\varphi} = \omega \end{cases} \tag{1}$$

where $\omega \in R$ represents the angular velocity. $p, q, s \in R$ are defined as linear velocities in surge direction and sway direction, respectively. From this above equation, we can derive the kinetic and potential energies of the hovercraft to compute L'agrange L=T-V. Then we can apply Euler-Lagrange formulation on equation (2):

$$M(q)\dot{q} + C(q,\dot{q})q = \begin{bmatrix} F \\ T_q \\ 0 \end{bmatrix} \quad (2)$$

where $T_q \in R$ represents the torque in yaw and $F \in R$ is the control force in the surge direction. The torque control, which is perpendicular from the center of the hovercraft propeller, is a function of $F \in R$.

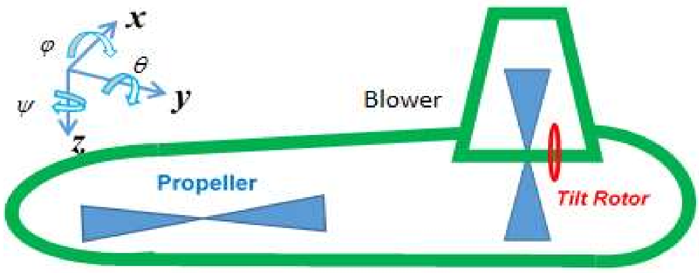


FIGURE 1. The Hovercraft prototype model.

3. Optimization Algorithms and Control Strategy

3.1.1. PD (Proportional and Derivative) *controller*

The PD control [32], which implemented in this article, is high efficiency despite owns a simplistic structure. Its design can reduce the settling time whereas improve the system stability. For the ith area, the PD controller combined gain, $K(s)$, is computed from the following equation (3):

$$G(s) = K_P + K_D.s \quad (3)$$

Where, K_P and K_D are the conventional PD controller proportional and derivative gains, respectively.

3.1.2. Fuzzy Logic Controller (FLC)

The FLC dynamic behavior are based on a set of linguistic rules and originated from expert knowledge [7-9]. Designer needs to decide the input and out variables for building a suitable set of Fuzzy rules. In this article, the error *e(t)* and the error rate *de(t)/dt* are chosen as input variables, while the output is considered as c_i. Then, the relationship between these two inputs and one output variables are investigated. The error and error rate as well as the Fuzzy rules determine the online change of the system output c_i. We then need to fuzzify and defuzzify *e(t), de(t)/dt and* c_i parameters. Fig. 2 illustrate the Fuzzy Inference System employed in this work. The center of area method (COA) defuzzification and Mandani's MIN–MAX inference engine is utilized. Seven linguistic triangular membership functions for the two inputs and one output are assigned: positive big PB (3), positive medium PM (2), positive small PS (1), zero ZE (0), negative small NS (-1), negative medium NM (-2), and negative big NB (-3). Table 1 explains the Fuzzy controller rules.

3.1.3 PD like Fuzzy controller

An intelligent control apply to autonomous hovercraft is the PD like Fuzzy controller. In that, FLC could be used as a classical PD (Proportional-Derivative) controller, and overcome the disadvantages of the PD-Controller. It is necessary to select the input and output variables and proper controller rules, as illustrated on equation (4)

$$u(t) = K_P * e(t) + K_D * de(t) \quad (4)$$

Where K_P and K_D are proportional gains and differential gain which called as the scaling factors, *e(t)* is the error and *de(t)* is the change in error. Due to optimize the Scaling Factors (SF) as the inputs of PD like Fuzzy controller, the proposed optimization algorithms are presented on the next sub-sections.

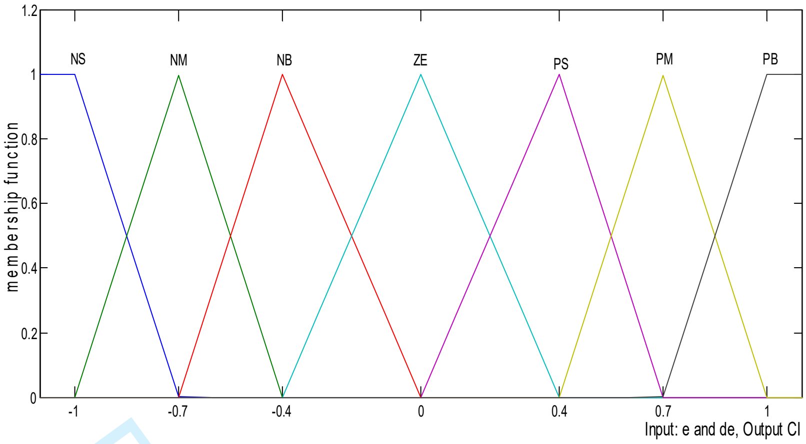


FIGURE 2. Fuzzy membership function.

TABLE 1. Rule base of Fuzzy Logic Controllers

Control Output CO (t)		***Input e(t)***						
		-3	**-2**	**-1**	**0**	**1**	**2**	**3**
Input de(t)	**-3**	**0**	1	2	2	**3**	**3**	**3**
	-2	-1	**0**	1	2	2	**3**	**3**
	-1	-2	-1	**0**	1	2	2	**3**
	0	-2	-2	-1	**0**	1	2	2
	1	**-3**	-2	-2	-1	**0**	1	2
	2	**-3**	**-3**	-2	-2	-1	**0**	1
	3	**-3**	**-3**	**-3**	-2	-2	-1	**0**

3.2.1. PSO Algorithm

Recently, the PSO has emerged as the one of the most well-known and powerful tools for optimization [18-22]. We try to adapt and improve PSO algorithm for the hovercraft control. Specifically, PSO algorithm is combined with another effective optimization algorithm to optimally tune the PD controller gains. The PSO implementation process is illustrated by the flow chart in Fig. 3. The PSO mathematical equations (5) and equation (6) are briefly described as follows:

$$\boldsymbol{V(t+1) = \omega V(t) + c_1 r_1 (Pbest(t) - P(t)) + c_2 r_2 (Gbest(t) - P(t))} \tag{5}$$

$$\boldsymbol{P(t+1) = P(t) + V(t+1)} \tag{6}$$

Where V is particle velocity, P is the current position, Pbest is local best position while Gbest is global best position; ω is the inertia weighting factor, c_1 and c_2 as learning rates. The variables r_1 and r_2 are random distribution values ϵ [0 - 1].

3.2.2. Bacterial Foraging (BF) optimization

The BF **optimization** derives from the searching foraging **of** the Escherichia **(E-coli)** bacteria [23, 33-36] capabilities to survival in the natural changing environment. The fitness criteria of evolution process depend on their motile behavior to maintain the good foraging strategy as well as reshapes or even eliminate the poor strategy when they are on the path to finding the food source. The bacteria genes with the good foraging strategy are then proliferated in the evaluation chain and reproduced the better bacteria in next generations. The E-coli bacteria foraging progression to global searching capability is simplify described by four significant steps of Chemo-tactic (θ_k), Swarming, Reproduction (N_r) and Elimination–Dispersal. First, Chemo-tactic (θ_k) process illustrates the E-coli bacteria motion by two different methods; one can swim to **fixed** time **while** the other can tumble to alternates between two operation of the lifetime modes. Where, θ_k **symbolizes** the k^{th} bacterium, N_k is the size of the step taken in the random direction specified by the tumble and Δ is a length of **the** random direction unit vector, as shown in equation (7). Second, Swarming step: bacteria move to follow the nutrient gradient produced by the nutrient consuming **group**. Bacteria will release an attractant when they reach high nutrient areas and form concentric patterns of higher density bacteria. Third, Reproduction (N_r) step**: t**o maintain the constant population, the healthier bacteria split into two and the less healthy bacteria die. As shown in equation (8), Sr is the number of population members that have had enough nutrients to reproduce (split in two) without mutation. Finally, Elimination and Dispersal step: may occur when local significant increases in heat kill bacteria that are currently in nutrient-rich regions. Bacteria can be dispersed by an abrupt flow of water. Elimination and dispersal events may destroy chemotactic progress, but they may also promote chemotaxis, since they place bacteria near food sources. Soon after BF algorithm was invented, it has been widely used for optimization in numerous science and technology fields.

$$\boldsymbol{\theta}_{\mathbf{k}} = \boldsymbol{\theta}_{\mathbf{k}} + \mathbf{N}_{\mathbf{k}} * \boldsymbol{\Delta}_{\mathbf{k}} \quad (7)$$

$$Sr = S/2 \quad (8)$$

3.2.3. Fusion Bacterial Foraging-Particle Swarm Optimization algorithm

Several studies [33-36] combined the ability of PSO to exchange social information with the ability of BF to eliminate and disperse old problems into new solutions**.** This combination was called as fusion BF-PSO, which has the combined strength of the two algorithms. To achieve optimal convergence, the proposed BF-PSO algorithm is derived from the specified search directions of bacteria tumbling, which are oriented simultaneously by the individual and global best locations, and the identified ideas of PSO is updating position and velocity. Several parameters must be evaluated, including iterations, inertia weight, learning rates, the number of bacteria, chemotactic steps, limit swimming distance, reproduction steps, elimination-dispersal events, and probability that each bacteria is eliminated are refered from [17-20, 23, 33-36] and the own experimental controller designs. Hence, we combine BF-PSO and the PD like Fuzzy control for building the hovercraft controller. This algorithm is then implemented to find the Scaling Factor (SF) as the best optimization parameters of the proposed controller. In the control diagram illustrated in Fig. 3, the tracking error e(t) and differential tracking error de(t) are employed as inputs of the Fuzzy Inference System (FIS). Two main parameters K_P (Proportional gain) and K_D (Derivative gain) of the PD like Fuzzy control are then optimally tuned with the Integral of Time Multiplied Square Error (ITSE) performance index [29-31]. After this, the results are compared qualitatively and quantitatively in terms of the ITSE minimum value: the settling time, the rising time, the overshoot and the tracking error, as demonstrated in equation (9):

$$\boldsymbol{ITSE} = \int_{\mathbf{0}}^{\infty} \boldsymbol{e}^{\mathbf{2}}(\boldsymbol{t}).\boldsymbol{t}.\boldsymbol{dt} \quad (9)$$

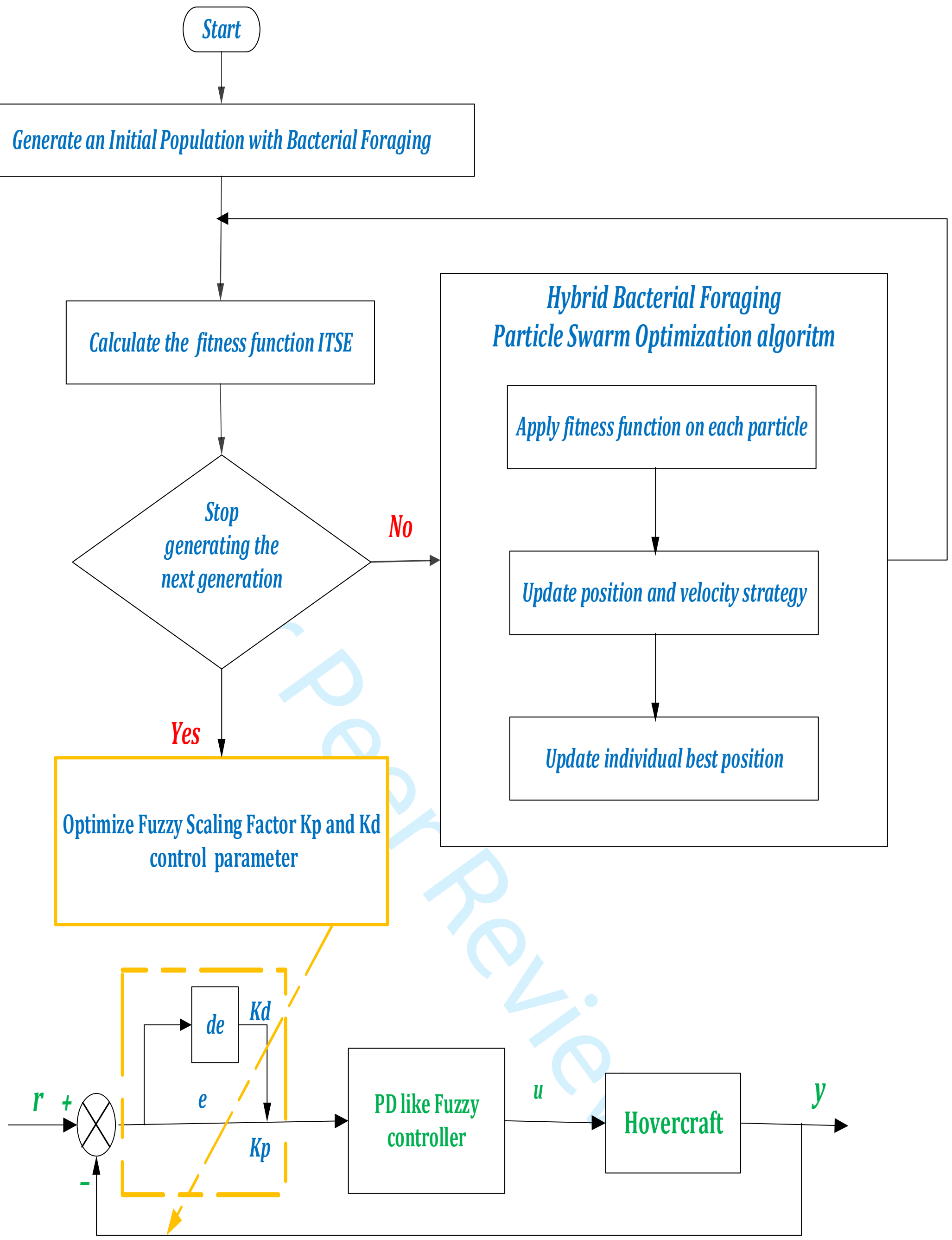


Figure 3. Fusion BF-PSO algorithm diagram

4. Numerical Peformance

The autonomous Hovercraft models were derived from the previous researches [1, 6 and 13] with the mass of Hovercraft $m = 2.1\ kg$ and the inertia moment $I = 0.000257$. Numerical simulation results are operated in the stochastic optimization space. The proposed optimizations are simply set on 50 iterations. The PD gains of the proposed controllers are set in range ϵ [0, 100]. The significant optimal parameters of each optimization algorithm are chosen and displayed in the Table 2.

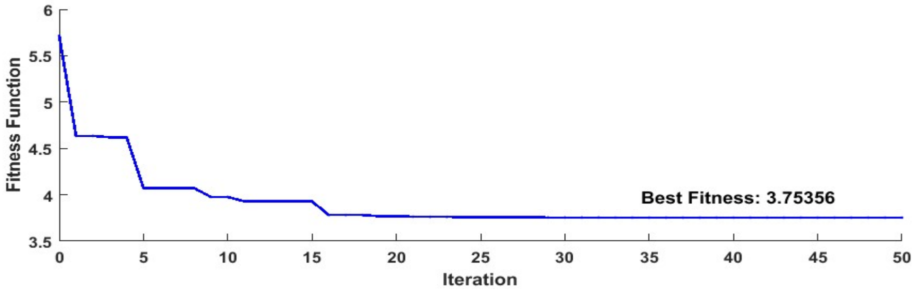
TABLE 2. Algorithms parameters setup

Parameters	PSO	BF	BF-PSO
Dimension of search space	2	2	2
Generation or Iteration	50	50	50
Acceleration coefficient, $c1$	1.22	-	1.22
Acceleration coefficient, $c2$	1.22	-	1.22
Inertia weighting factor	0.92	-	0.92
Number of bacteria	-	50	50
Number of chemotactic steps	-	10	10
Limits the length of a swim	-	5	5
Number of reproduction steps	-	5	5
Number of elimination-dispersal events	-	3	3
Probability of each bacteria getting eliminated or dispersed	-	0.2	0.2

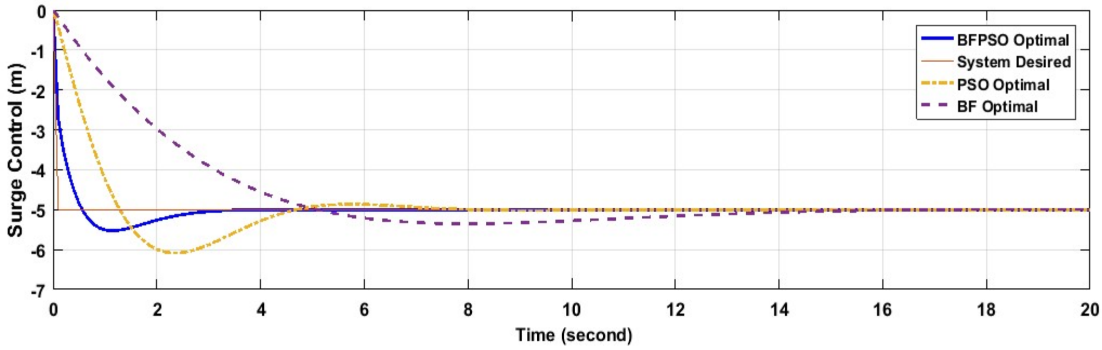
For the purpose of comparing the fusion BFPSO with the standard BF and conventional PSO methods in shorter generations of updating the fitness cost function, this paper investigate the performances of three channels of autonomous Hovercraft motions: Surge position x in Figure 4, Sway position y in Figure 5 and yaw Steering angle in Figure 6, respectively. The numerical performances display that the fusion BF-PSO methodology achieved the significant stability and dominance in performance response compare to the solo BF or the single PSO algorithm. All presented results proved that BFPSO achieved the optimal control including: response time, less error, just after the third second at all, and were illustrated on the Table 3. Generally, all of the optimal controller designs are achieved the high-quality performance in terms of stability, high precision and high reliability.

TABLE 3. The BF-PSO proposed method tuning gain results

	Surge position x (5 m)	***Sway position y (5 m)***	***Steering-Yaw angle (10 degree)***
Generation	*50*	*50*	*50*
K_P	*78.324*	*45.783*	*34.569*
K_D	*50.671*	*88.123*	*77.257*
Settling time (s)	*2.943*	*2.818*	*3.154*
Overshoot (%)	*10.12*	*1.05*	*0.93*
Elapsed time (s)	***21009.3397***	***29053.3585***	***25266.4916***

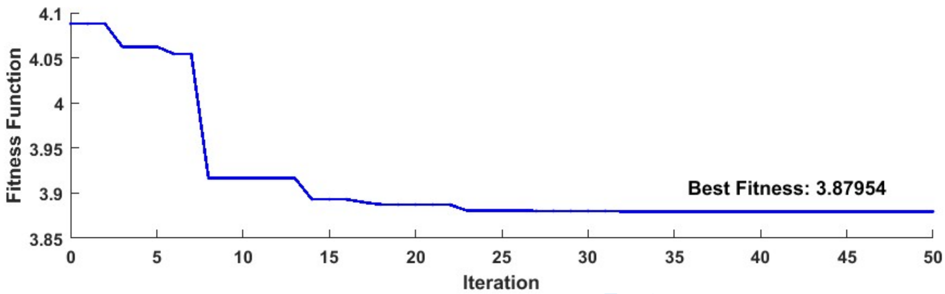


(a)

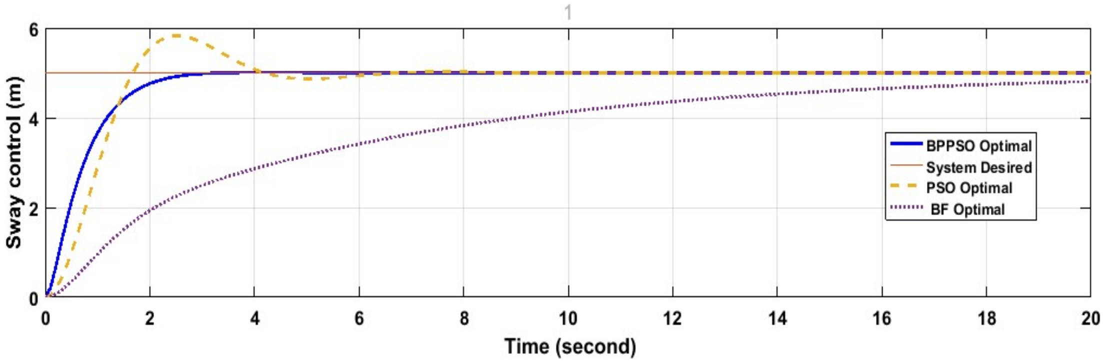


(b)

FIGURE 4. Hovercraft Surge control. (a) Fitness value, (b) The proposed algorithm.

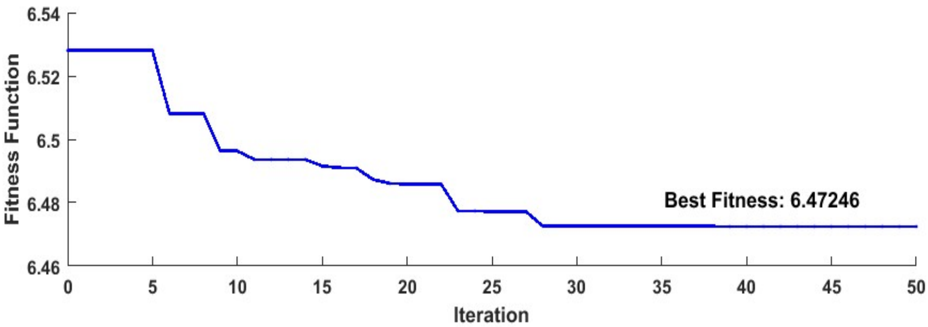


(a)

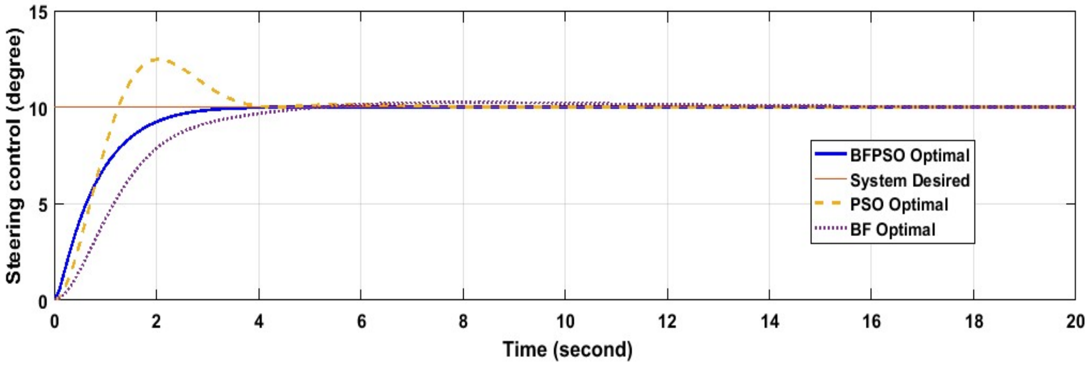


(b)

FIGURE 5. Hovercraft Sway control. (a) Fitness value, (b) The proposed algorithm.



(a)



(b)

FIGURE 6. Hovercraft Steering angle control. (a) Fitness value, (b) The proposed algorithm.

5. Conclusions

Three numeric optimal controllers design has been successfully implemented for pilot an autonomous Hovercraft in this paper. The fusion BF-PSO algorithm, in order to employ the ability to exchange social information of PSO algorithm also the ability to find the new solution of BF process via the elimination and dispersal step, is proposed to find the optimal Fuzzy- like PD controller parameters; it is then embedded to the hovercraft pilot models: surge, sway and steering control. The proposed controller obtains the optimal pilot performance after the third second when compared with the other single optimal controller designs, PSO or BF. Furthermore, it provides fastest response, high reliability as well as stability than available controllers of the realistic hovercraft mathematical models. In further work, the method should be modified to reduce the unwanted disturbances attacking to the system model. Besides, the real hovercraft model with proposed control technique will be implemented for testing and optimization.

References

1. D. Cabecinhas, P. Batista, P. Oliveira and C. Silvestre, "Hovercraft Control with Dynamic Parameters Identification," in IEEE Transactions on Control Systems Technology, vol. 26, no. 3, pp. 785-796, May 2018, doi: 10.1109/TCST.2017.2692733.
2. Hebertt Sira-Ramirez and Carlos Aguilar Ibanez, “On the control of the Hovercraft”. Dynamic and Control, Vol. 10, pp. 151-163, Kluwer Academic Publishers, 2000.
3. Silvano Balemi, Robert Bucher, Paola Guggiari, Ivan Furlan, Markus Kottmann and Jacques Chapuis, “Rapid control of prototyping of a Hovercraft”, Conference: Proc. of MSy'02, Embedded Systems Conference, October, 2002.

4. Rashid, Aras, Kassim, Ibrahim, Jamali, “Dynamic Mathematical Modeling and Simulation Study of Small-scale Autonomous Hovercraft”, International Journal of Advanced Science and Technology, Vol. 46, September, 2012.
5. Dictino Chaos, David Moreno-Salinas, Rocío Muñoz-Mansilla, and Joaquín Aranda, “Nonlinear Control for Trajectory Tracking of a Nonholonomic RC-Hovercraft with Discrete Inputs,” Mathematical Problems in Engineering, vol. 2013, Article ID 589267, 16 pages, 2013.
6. Garcia D. I. and White W. N., “Control design of an unmanned hovercraft for agricultural applications,” International Journal of Agricultural and Biological Engineering, vol. 8, no. 2, pp. 72–79, 2015.
7. E. H. Mamdani, “Application of fuzzy algorithms for control of simple dynamic plant,” Proceedings of the Institution of Electrical Engineers, vol. 121, no. 12, pp. 1585–1588, 1974.
8. E. H. Mamdani, “Application of fuzzy logic to approximate reasoning using linguistic synthesis,” IEEE Transactions on Computers, vol. C-26, no. 12, pp. 1182–1191, 1977.
9. Z.-Y. Zhao, M. Tomizuka, and S. Isaka, “Fuzzy gain scheduling of PID controllers,” IEEE Transactions on Systems, Man and Cybernetics, vol. 23, no. 5, pp. 1392–1398, 1993.
10. Sanchez E.N., Becerra H.M. and Velez C.M. (2007) “Combining fuzzy, PID and regulation control for an autonomous mini-helicopter”, Information Sciences 177, pp. 1999–2022.
11. Precup R.E. and Preitl S. (2007) PI-Fuzzy controllers for integral plants to ensure robust stability, Information Sciences 177, pp. 4410–4429.
12. Juang Y.T., Chang Y.T. and Huang C.P. (2008) “Design of fuzzy PID controllers using modified triangular membership functions”, Information Sciences 178, pp. 1325–1333.
13. Tran, H.K.; Son, H.H.; Duc, P.V.; Trang, T.T.; Nguyen, H.-N. Improved Genetic Algorithm Tuning Controller Design for Autonomous Hovercraft. Processes 2020, 8, 66. https://doi.org/10.3390/pr8010066
14. Tran, H.K.; Chiou, J.-S.; Dang, V.-H. New Fusion Algorithm-Reinforced Pilot Control for an Agricultural Tricopter UAV. Mathematics 2020, 8, 1499. https://doi.org/10.3390/math8091499
15. Passino K.M. and Yurkovich S. (1998) Fuzzy Control, Addison-Wesley, Reading, MA.
16. Eiben, A.E.; Smith, J.E. Introduction to Evolutionary Computing; Springer: Berlin, Germany, 2003.
17. Yang, Xin-She, Engineering Optimization: An Introduction with Metaheuristic Applications, John Wiley&Sons, Inc. University of Cambridge, 2010.
18. Kennedy J. and Eberhart R.C. (1995) “Particle swarm optimization”, Proceedings IEEE International Conference on Neural Networks IV pp. 1942–1948.
19. Clerc M. (1999) “The swarm and the queen: Towards a deterministic and adaptive particle swarm optimization”, Proceedings of the ICEC, Washington, DC, pp. 1951–1957.
20. Eberhart R.C. and Shi Y (2000) “Comparing inertia weights and constriction factors in particle swarm optimization”, Proc. of the IEEE Congress on Evolutionary Computation, USA pp. 84–88.
21. Bergh F. and Engelbrecht A.P. (2004) "A Cooperative approach to particle swarm optimization," IEEE Transactions on Evolutionary Computation, vol. 8, no. 3, pp. 225-239.
22. Huang T and Mohan AS (2005) “A hybrid boundary condition for robust particle swarm optimization”, IEEE Antennas and Wireless Propagation Letters 4, pp. 112–117.
23. Passino,K.M., “Biomimicry of bacterial foraging for distributed optimization and control,” IEEE Control Syst. Mag. 22 (2002) 52–67.
24. Juang C.F. (2004) “A hybrid of genetic algorithm and particle swarm optimization for recurrent network design”, IEEE System, Man, and Cybernetics: B 34 (2), pp. 997–1006.

25. Li L.L., Wang L. and Liu L.L. (2006), “An effective hybrid PSOSA strategy for optimization and its application to parameter estimation”, Applied Mathematics and Computation 179, pp. 135–146.
26. Ali A.F. and Tawhid M.A. (2016) “A Hybrid PSO and DE Algorithm for Solving Engineering Optimization Problems”, Appl. Math. Inf. Sci. 10, No. 2, pp. 431-449.
27. Chen CL, Lin YL, Feng YC (2018) "Optimization of large-scale economic dispatch with valve-point effects using a modified hybrid PSO-DSM approach", Journal of Marine Science and Technology. DOI: 10.6119/JMST-015-1215-1
28. Hsu CI, Wu SPJ, Chiu CC (2019), "A Hybrid Swarm Intelligence Approach for Blog Success Prediction", International Journal of Computational Intelligence Systems, Vol. 12, Iss. 2, pp. 571 – 579
29. Skogestad, S. “Simple analytic rules for model reduction and PID controller tuning”, Journal of Process Control, Vol. 13, pp. 291-309, 2003.
30. Martins, F. G. “Tuning PID Controllers using the ITAE Criterion” International Journal Engineering Education, 21(5), pp. 867-873, 2005.
31. Tan, W.; Liu, J.; Chen, T.; Marquez, H.J. “Comparison of some well-known PID tuning formulas”. Computation Chemical Engineering, pp. 1416–1423, 2006.
32. Tran, HK, Chiou, JS, Peng, ST. Design genetic algorithm optimization education software based fuzzy controller for a tricopter fly path planning. EURASIA Journal of Mathematics, Science and Technology Education 2016; 12(5): 1303–12.
33. W. M. Korani, H. T. Dorrah and H. M. Emara, "Bacterial foraging oriented by Particle Swarm Optimization strategy for PID tuning," 2009 IEEE International Symposium on Computational Intelligence in Robotics and Automation - (CIRA), 2009, pp. 445-450, doi: 10.1109/CIRA.2009.5423165.
34. Liu XiaoLong, Li RongJun and YangPing, "A bacterial foraging global optimization algorithm based on the particle swarm optimization," 2010 IEEE International Conference on Intelligent Computing and Intelligent Systems, 2010, pp. 22-27, doi: 10.1109/ICICISYS.2010.5658828.
35. S.M. Abd-Elazim, E.S. Ali, “A hybrid Particle Swarm Optimization and Bacterial Foraging for optimal Power System Stabilizers design,” International Journal of Electrical Power & Energy Systems, Volume 46, 2013, Pages 334-341, https://doi.org/10.1016/j.ijepes.2012.10.047.
36. Kora, P., Kalva, S.R. Hybrid Bacterial Foraging and Particle Swarm Optimization for detecting Bundle Branch Block. Springer Plus 4, 481 (2015). https://doi.org/10.1186/s40064-015-1240-z

Fusion Bacterial Foraging and Particle Swarm Optimization Algorithm For An Autonomous Hovercraft Control

Huu Khoa Tran[1], Pham Duc Lam[2], Tran Thanh Trang[3], and Hoang Nam Nguyen[4,*]

[1] Department of Information Management, Tamkang University, New Taipei City, Taiwan;
email: hk5tran@mail.tku.edu.tw, https://orcid.org/0000-0001-5268-2355
[2] Faculty of Engineering and Technology, Nguyen Tat Thanh University, Vietnam
email: pdlam@nttu.edu.vn
[3] Faculty of Electrical and Electronics Engineering, Ho Chi Minh City University of Food Industry, Ho Chi Minh City, Vietnam; trangtt@hufi.edu.vn
[4*] Modeling Evolutionary Algorithms Simulation and Artificial Intelligence, Faculty of Electrical and Electronics Engineering, Ton Duc Thang University, 19 Nguyen Huu Tho Street, Tan Phong Ward, District 7, Ho Chi Minh City, Vietnam;
* Correspondence: nguyenhoangnam@tdtu.edu.vn.

Abstract: This article proposes a novel method for designing an effective autonomous hovercraft vehicle controller. In the first step, we design a PD like Fuzzy control, in which Proportional and Derivative (PD) parameters are used as the Fuzzy Scaling Factor (FSF) of the controller design. Next, Particle Swarm Optimization (PSO) improved by fusion Bacterial Foraging (BF) (named hereafter fBF-PSO) is proposed to optimize the proportional and derivative gains of the PD like Fuzzy controller. The integral of time multiplied square error (ITSE) fitness function is applied to find the minima of the controller design error criterion. Numerical simulations are then implemented on a real autonomous hovercraft motion model. Results indicate that the proposed controller design is more stable, reliable and accurate than existing controllers for the real hovercraft mathematical model.

Keywords: Bacterial Foraging (BF) Algorithm, Particle Swarm Optimization (PSO), PD like Fuzzy, Hovercraft, Integral of Time Multiplied Square Error (ITSE).

1. Introduction

In recent years, many researchers have focused on using nonlinear control to build an autonomous hovercraft [1–6] that can work well on various surfaces such as water, sand beach, ice and especially on muddy land. Most of these available controllers of the real hovercraft models are just based on the traditional PD controller and very few methods of optimizing the controller parameters are introduced for this system. The main aim of this paper is to propose a new way to optimize the proportional and derivative gains of the PD controller of the hovercraft.

Based on expert knowledge system, the Fuzzy Logic Controllers (FLC) have the advantage solution to the issues that could realize by human operators. Many works [7-15] has utilize this technique to solve various control system subjects and obtain considerably better results compare to others. When physical process has uncertain information and limited data, Fuzzy rule-based model is an appropriate choice. Therefore, we also use this approach in the hovercraft model control system. In this paper, we design a set of Fuzzy triangular membership functions and optimizes Scaling Factors (SF) as the inputs of Fuzzy controller. Although evolutionary algorithms [16, 17] such as Particle Swarm Optimization (PSO), Simulated Annealing (SA) and Genetic Algorithm (GA), can easily detect the local minima but it is difficult to detect the global optimization. Particle swarm optimization (PSO) [18-22] is a population-based heuristic algorithm that is invented based on observing animal swarm social behavior to detect accurate targets in multi-dimensional space. PSO uses particles (individuals), which are iteratively updated in each iteration, to perform searches. For finding the optimal solution, each particle decides its search direction based on its best previous location (cognitive part) and all other members' best locations (social part). In 2002, K. M. Passino [23], inspired from the Bacteria Foraging (BF) behavior of *Escherichia-coli* bacteria, firstly proposed a bionic algorithm that is very effective for distributed optimization and control design. Although, BF algorithm is very effective in local search but it is easily tapped or delayed in reaching global solution because of the random search directions during chemotaxis process. Due to eliminating the local optima trap and improving the optimal process, the modern engineering application researchers have introduced various PSO-based algorithms to solve the problems, for instance: GA and PSO [24], PSO and SA [25], hybrid PSO-Differential Evolution [26], direct search method (DSM) and hybrid PSO [27], hybrid Swam Intelligence approach [28]. Also, combining PSO with bacterial foraging has emerged as one of the most efficient optimization methods, which can be applied successfully in many fields, including mathematical optimization [29], RFID network planning [30], medical care [31] and traditional PID control [32-37] in electric power systems. The above techniques can expand the optimal convergence pace by taking the advantages while eliminating the weakness of the solo algorithms.

Thus, we developed a fusion algorithm, called fusion Bacterial Foraging and Particle Swarm Optimization (fBF-PSO), to increase the convergence speed in shorter operating generation and adjust the global search efficiency. We use PD control to improve hovercraft system stability. Additionally, a Fuzzy Inference System (FIS), that takes advantage of PID conventional control, is utilized for reducing the steady state error. The fusion BF-PSO technique is also used to improve the performance of this PD like Fuzzy control by minimizing the ITSE fitness cost function [38-41]. The proposed controller can obtain the high-performance control of the Hovercraft in terms of higher accuracy, quick response, high stability and smoother maneuver. To the best of our knowledge, this is the first paper that combines the advantages of PSO and BF method in designing an effective and optimal PD controller for a real hovercraft model. In section 2, we describe the hovercraft configuration. In section 3, optimization algorithm and control strategy are introduced. In section 4, numerical simulation control of a real hovercraft model is presented. Finally, conclusions are discussed in section 5.

2. Autonomous Hovercraft configuration

The Air Cushion Vehicle (ACV), which is usually called as hovercraft, is composed of rotors and a cushion. Blower (rotor duct fan) is used to provide a large volume of air inside the air cushion, which is larger than the atmospheric pressure. Therefore, the hovercraft can float and is capable of travelling over ice, water, land, mud… [1-6, 13]. A tilt servo motor, which is located at the rear, is usually employed to steer the hovercraft. Although many modern technologies are utilized, there is still needs for a more advanced hovercraft maneuvering system with better performance. Especially, faster response and higher reliability are of great interest. Figure 1 shows a popular hovercraft model used in this research, which has a single tilt servo motor, a propeller settled along z-axis and a blower attached along y-axis.

The dynamic Hovercraft model utilized in this paper is referenced from [1, 6 and 13] with right hand coordinate systems. X-axis is used to position the lateral direction for controlling the sway motion and surge position. The Z-axis is in vertical direction and its positive direction is downwards. Whilst Y-axis is along its body, which is necessary for controlling sway motion or surge position. The hovercraft's kinematics can be described as equation (1):

$$\begin{cases} \dot{x} = p\cos\varphi - s\sin\varphi \\ \dot{y} = q\cos\varphi - s\sin\varphi \\ \dot{\varphi} = \omega \end{cases} \tag{1}$$

where $\boldsymbol{\omega} \in \boldsymbol{R}$ represents the angular velocity. $\boldsymbol{p}, \boldsymbol{q}, \boldsymbol{s} \in \boldsymbol{R}$ are defined as linear velocities in surge direction, sway direction and steering, respectively. From this above equation, we can derive the kinetic and potential energies of the hovercraft to compute L'agrange $L=T\text{-}V$. Then we can apply Euler-Lagrange formulation on equation (2):

$$\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\boldsymbol{q} = \begin{bmatrix} \boldsymbol{F} \\ \boldsymbol{T_q} \\ \boldsymbol{0} \end{bmatrix} \quad (2)$$

where $\boldsymbol{T_q} \in \boldsymbol{R}$ represents the torque in yaw and $\boldsymbol{F} \in \boldsymbol{R}$ is the control force in the surge direction. The torque control, which is perpendicular from the center of the hovercraft propeller, is a function of $\boldsymbol{F} \in \boldsymbol{R}$.

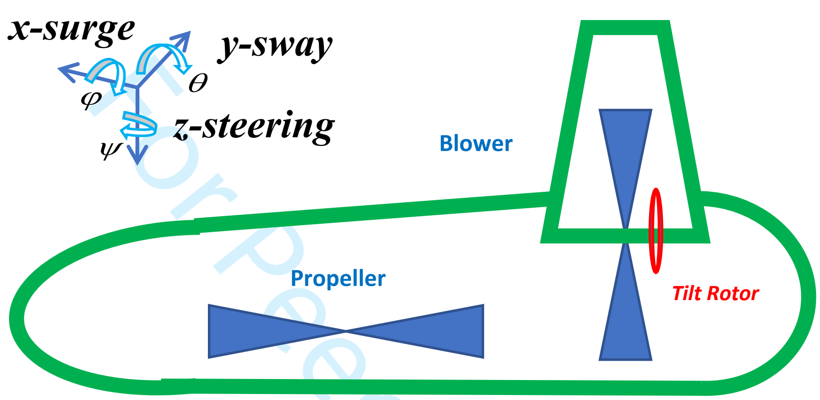


Figure 1. The Hovercraft prototype model.

3. Optimization Algorithms and Control Strategy

3.1.1. PD (Proportional and Derivative) *controller*

The PD control, which implemented in this article, is high efficiency despite owns a simplistic structure. Its design can reduce the settling time whereas improve the system stability. For the ith area, the PD controller combined gain, $K(s)$, is computed from the following equation (3):

$$\boldsymbol{G}(\boldsymbol{s}) = \boldsymbol{K_P} + \boldsymbol{K_D}.\boldsymbol{s} \quad (3)$$

Where, K_P and K_D are the conventional PD controller proportional and derivative gains, respectively.

3.1.2. Fuzzy Logic Controller (FLC)

The FLC dynamic behavior are based on a set of linguistic rules and originated from expert knowledge [7-9]. Designer needs to decide the input and out variables for building a suitable set of Fuzzy rules. In this article, the error *e(t)* and the error rate *de(t)/dt* are chosen as input variables, while the output is considered as c_i. Then, the relationship between these two inputs and one output variables are investigated. The error and error rate as well as the Fuzzy rules determine the online change of the system output c_i. We then need to fuzzify and defuzzify *e(t), de(t)/dt and* c_i parameters. Fig. 2 illustrate the Fuzzy Inference System employed in this work. The center of area method (COA) defuzzification and Mandani's MIN–MAX inference engine is utilized. Seven linguistic triangular membership functions for the two inputs and one output are assigned: positive big PB (3), positive medium PM (2), positive small PS (1), zero ZE (0), negative small NS (-1), negative medium NM (-2), and negative big NB (-3). Table 1 explains the Fuzzy controller rules.

3.1.3 PD like Fuzzy controller

An intelligent control apply to autonomous hovercraft is the PD like Fuzzy controller. In that, FLC could be used as a classical PD (Proportional-Derivative) controller, and overcome the disadvantages of the PD-Controller. It is necessary to select the input and output variables and proper controller rules, as illustrated on equation (4)

$$\boldsymbol{u(t) = K_P.e(t) + K_D.de(t)} \quad (4)$$

Where K_P and K_D are proportional gains and differential gain which called as the scaling factors, *e(t)* is the error and *de(t)* is the change in error. Due to optimize the Scaling Factors (SF) as the inputs of PD like Fuzzy controller, the proposed optimization algorithms are presented on the next sub-sections.

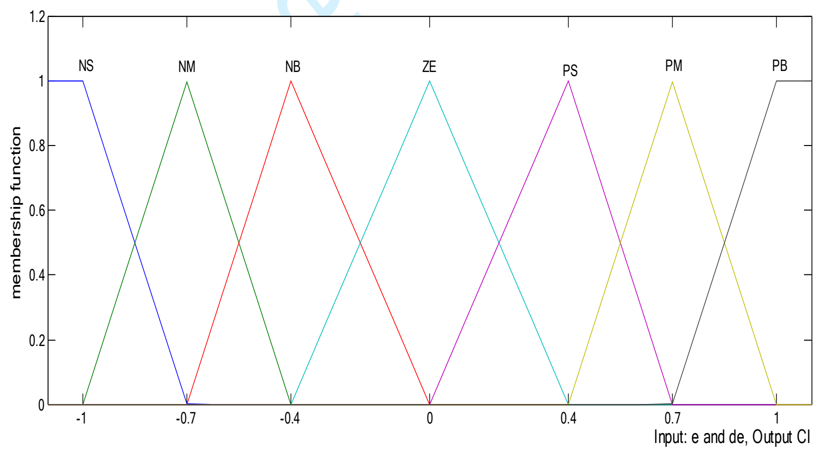


Figure 2. Fuzzy membership function.

Table 1. Rule base of Fuzzy Logic Controllers

Control Output CI (t)		Input e(t)						
		-3	-2	-1	0	1	2	3
Input de(t)	-3	0	1	2	2	3	3	3
	-2	-1	0	1	2	2	3	3
	-1	-2	-1	0	1	2	2	3
	0	-2	-2	-1	0	1	2	2
	1	-3	-2	-2	-1	0	1	2
	2	-3	-3	-2	-2	-1	0	1
	3	-3	-3	-3	-2	-2	-1	0

3.2.1. PSO Algorithm

Recently, the PSO has emerged as the one of the most well-known and powerful tools for optimization [18-22]. We try to adapt and improve PSO algorithm for the hovercraft control. Specifically, PSO algorithm is combined with another effective optimization algorithm to optimally tune the PD controller gains. The PSO implementation process is illustrated by the flow chart in Fig. 3. The PSO mathematical equations (5) and equation (6) are briefly described as follows:

$$\boldsymbol{V(t+1) = \omega V(t) + c_1 r_1 (Pbest(t) - P(t)) + c_2 r_2 (Gbest(t) - P(t))} \quad (5)$$

$$\boldsymbol{P(t+1) = P(t) + V(t+1)} \quad (6)$$

Where V is particle velocity, P is the current position, Pbest is local best position while Gbest is global best position; ω is the inertia weighting factor, c_1 and c_2 as learning rates. The variables r_1 and r_2 are random distribution values ϵ [0 - 1].

3.2.2. Bacterial Foraging (BF) optimization

The BF optimization derives from the searching foraging of the Escherichia **(E-coli)** bacteria [23] capabilities to survival in the natural changing environment. The fitness criteria of evolution process depend on their motile behavior to maintain the good foraging strategy as well as reshapes or even eliminate the poor strategy when they are on the path to finding the food source. The bacteria genes with the good foraging strategy are then proliferated in the evaluation chain and reproduced the better bacteria in next generations. The E-coli bacteria foraging progression to global searching capability is simplify described by four significant steps of Chemo-tactic (θ_k), Swarming, Reproduction (N_r) and Elimination–Dispersal. First, Chemo-tactic (θ_k) process illustrates the E-coli bacteria motion by two different methods; one can swim to fixed time while the other can tumble to alternates between two operation of the lifetime modes. Where, θ_k

symbolizes the k^{th} bacterium, N_k is the size of the step taken in the random direction specified by the tumble and Δ is a length of the random direction unit vector, as shown in equation (7). Second, Swarming step: bacteria move to follow the nutrient gradient produced by the nutrient consuming group. Bacteria will release an attractant when they reach high nutrient areas and form concentric patterns of higher density bacteria. Third, Reproduction (N_r) step**: t**o maintain the constant population, the healthier bacteria split into two and the less healthy bacteria die. As shown in equation (8), Sr is the number of population members that have had enough nutrients to reproduce (split in two) without mutation. Finally, Elimination and Dispersal step: may occur when local significant increases in heat kill bacteria that are currently in nutrient-rich regions. Bacteria can be dispersed by an abrupt flow of water. Elimination and dispersal events may destroy chemotactic progress, but they may also promote chemotaxis, since they place bacteria near food sources. Soon after BF algorithm was invented, it has been widely used for optimization in numerous science and technology fields.

$$\boldsymbol{\theta_k = \theta_k + N_k} * \Delta_{\boldsymbol{k}} \quad (7)$$

$$\boldsymbol{S_r = S/2} \quad (8)$$

3.2.3. Fusion Bacterial Foraging-Particle Swarm Optimization algorithm

Several studies [29-37] combined the ability of PSO to exchange social information with the ability of BF to eliminate and disperse old problems into new solutions**.** This combination was called as fusion BF-PSO, which has the combined strength of the two algorithms. To achieve optimal convergence, the proposed fusion BF-PSO algorithm is derived from the specified search directions of bacteria tumbling, which are oriented simultaneously by the individual and global best locations, and the identified ideas of PSO is updating position and velocity. Several parameters must be evaluated, including iterations, inertia weight, learning rates, the number of bacteria, chemotactic steps, limit swimming distance, reproduction steps, elimination-dispersal events, and probability that each bacterium is eliminated are referred from [17-20, 23, 29-37] and the own experimental controller designs. Hence, we combine BF-PSO and the PD like Fuzzy control for building the hovercraft controller. This algorithm is then implemented to find the Scaling Factor (SF) as the best optimization parameters of the proposed controller. In the control diagram illustrated in Fig. 3, the tracking error e(t) and differential tracking error de(t) are employed as inputs of the Fuzzy Inference System (FIS). Two main parameters K_P (Proportional gain) and K_D (Derivative gain) of the PD like Fuzzy control are then optimally tuned with the Integral of Time Multiplied Square Error (ITSE) performance index [38-41]. After this, the results are compared qualitatively and quantitatively in terms of the ITSE minimum value: the settling time, the rising time, the overshoot and the tracking error, as demonstrated in equation (9):

$$\boldsymbol{ITSE = \int_0^\infty e^2(t).t.dt} \quad (9)$$

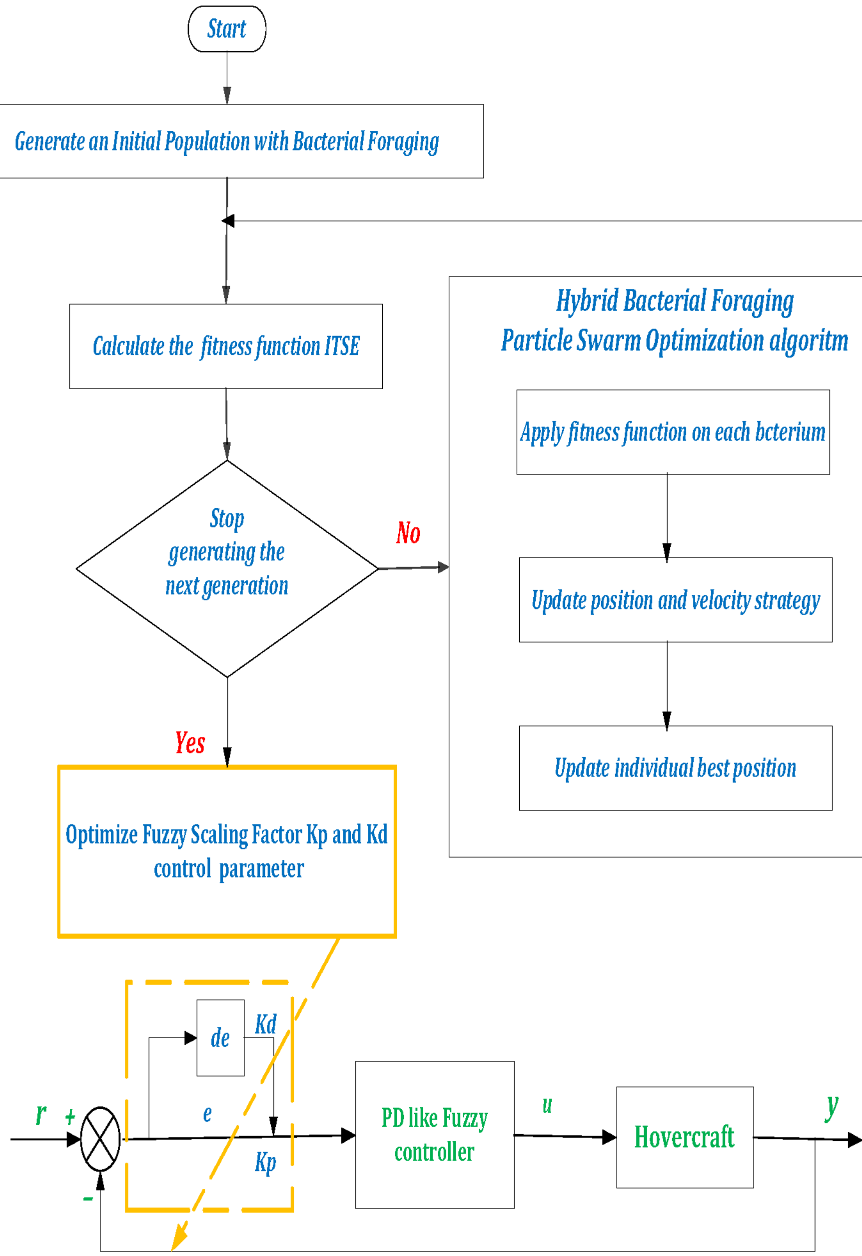


Figure 3. Fusion BF-PSO algorithm diagram

4. Numerical Peformance

The autonomous Hovercraft models were derived from the previous researches [1, 6 and 13] with the mass of Hovercraft $m = 2.1\ kg$ and the inertia moment $I = 0.000257$. Numerical simulation results are operated in the stochastic optimization space. The proposed optimizations are simply set on 50 iterations. The PD gains of the proposed controllers are set in range ϵ [0, 100]. The significant optimal parameters of each optimization algorithm are chosen and displayed in the Table 2.

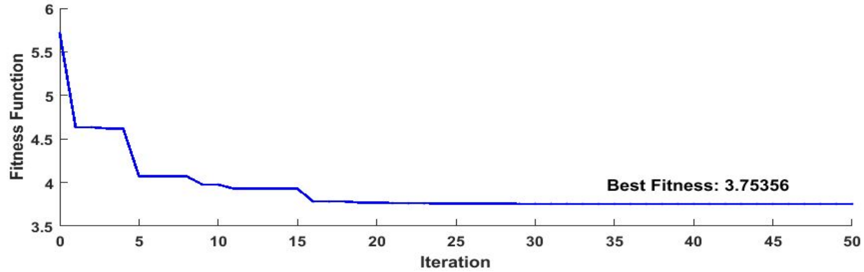
Table 2. Algorithms parameters setup

Parameters	PSO	BF	BF-PSO
Dimension of search space	2	2	2
Generation or Iteration	50	50	50
Acceleration coefficient, $c1$	1.22	-	1.22
Acceleration coefficient, $c2$	1.22	-	1.22
Inertia weighting factor	0.92	-	0.92
Number of bacteria	-	50	50
Number of chemotactic steps	-	10	10
Limits the length of a swim	-	5	5
Number of reproduction steps	-	5	5
Number of elimination-dispersal events	-	3	3
Probability of each bacteria getting eliminated or dispersed	-	0.2	0.2

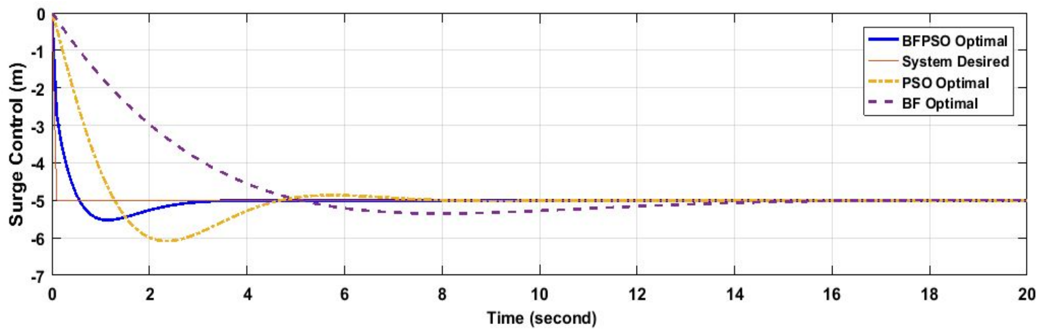
For the purpose of comparing the fusion BFPSO with the standard BF and conventional PSO methods in shorter generations of updating the fitness cost function, this paper investigate the performances of three channels of autonomous Hovercraft motions: Surge position x in Figure 4, Sway position y in Figure 5 and yaw Steering angle in Figure 6, respectively. The numerical performances display that the fusion BF-PSO methodology achieved the significant stability and dominance in performance response compare to the solo BF or the single PSO algorithm. All presented results proved that BFPSO achieved the optimal control including: response time, less error, just after the third second at all, and were illustrated on the Table 3. Generally, all of the optimal controller designs are achieved the high-quality performance in terms of stability, high precision and high reliability.

Table 3. The fusion BF-PSO proposed method tuning gain results

	Surge position x (5 m)	***Sway position y (5 m)***	***Steering-Yaw angle (10 degree)***
Generation	*50*	*50*	*50*
K_P	*78.324*	*45.783*	*34.569*
K_D	*50.671*	*88.123*	*77.257*
Settling time (s)	*2.943*	*2.818*	*3.154*
Overshoot (%)	*10.12*	*1.05*	*0.93*
Elapsed time (s)	***21009.3397***	***29053.3585***	***25266.4916***

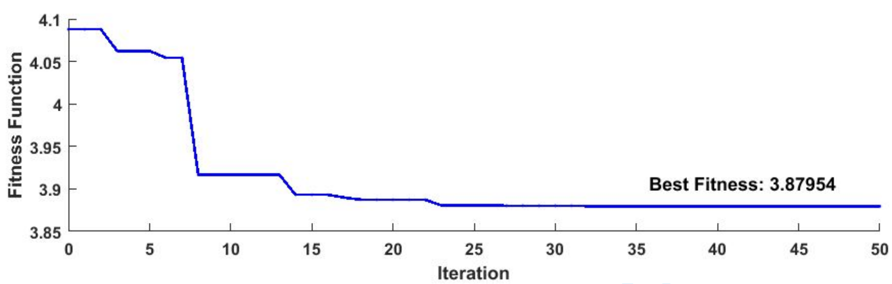


(a)

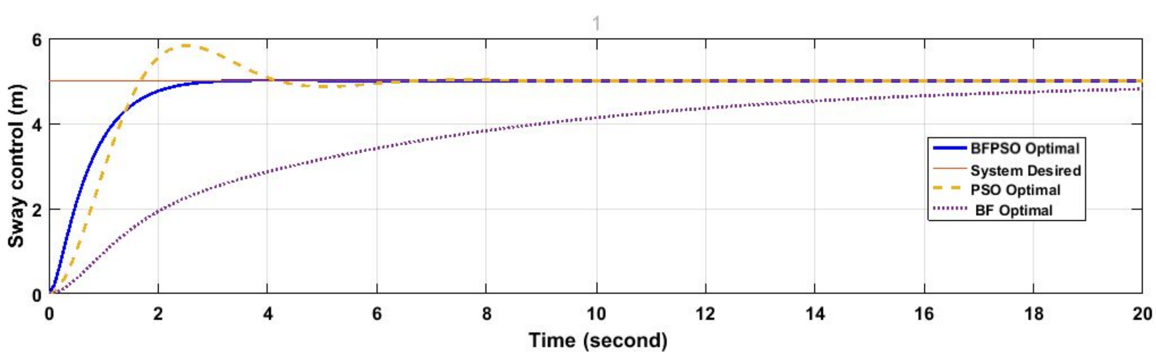


(b)

Figure 4. Hovercraft Surge control. (a) BFPSO algorithm fitness value, (b) The proposed algorithm.



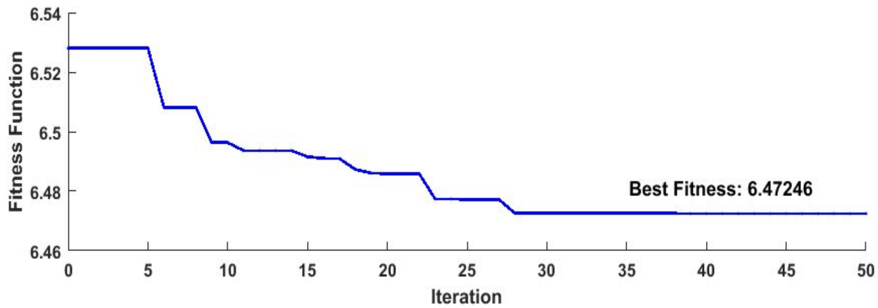
(a)



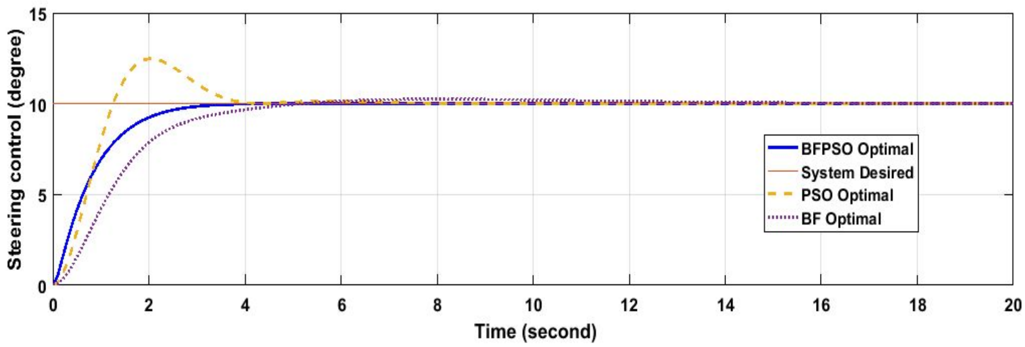
(b)

Figure 5. Hovercraft Sway control. (a) BFPSO algorithm fitness value, (b) The proposed algorithm

9



(a)



(b)

Figure 6. Hovercraft Steering angle control. (a) BFPSO algorithm fitness value, (b) The proposed algorithm.

5. Conclusions

The paper presents a successful design of three numerical optimal controllers to pilot an autonomous hovercraft. The fusion BF-PSO algorithm that employs both the ability to exchange social information of the PSO algorithm and the capability of finding new BF solutions via the elimination and dispersal step is proposed in order to find the optimal Fuzzy- like PD controller parameters for the hovercraft pilot models: surge, sway and steering control. The proposed controller obtains optimal pilot performance after the third second when compared with the other single optimal controller designs, PSO or BF. Furthermore, it provides fastest response, high reliability as well as stability than available controllers for the realistic hovercraft mathematical models. In future work, the approach method should be modified to reduce unwanted disturbances attacking the system model. Besides, the real hovercraft model with the proposed control technique will be implemented for testing and optimization.

References

1. D. Cabecinhas, P. Batista, P. Oliveira and C. Silvestre, "Hovercraft Control with Dynamic Parameters Identification," in IEEE Transactions on Control Systems Technology, vol. 26, no. 3, pp. 785-796, May 2018, doi: 10.1109/TCST.2017.2692733.
2. Hebertt Sira-Ramirez and Carlos Aguilar Ibanez, “On the control of the Hovercraft”. Dynamic and Control, Vol. 10, pp. 151-163, Kluwer Academic Publishers, 2000.

3. Silvano Balemi, Robert Bucher, Paola Guggiari, Ivan Furlan, Markus Kottmann and Jacques Chapuis, "Rapid control of prototyping of a Hovercraft", Conference: Proc. of MSy'02, Embedded Systems Conference, October, 2002.
4. Rashid, Aras, Kassim, Ibrahim, Jamali, "Dynamic Mathematical Modeling and Simulation Study of Small-scale Autonomous Hovercraft", International Journal of Advanced Science and Technology, Vol. 46, September, 2012.
5. Dictino Chaos, David Moreno-Salinas, Rocío Muñoz-Mansilla, and Joaquín Aranda, "Nonlinear Control for Trajectory Tracking of a Nonholonomic RC-Hovercraft with Discrete Inputs," Mathematical Problems in Engineering, vol. 2013, Article ID 589267, 16 pages, 2013.
6. Garcia D. I. and White W. N., "Control design of an unmanned hovercraft for agricultural applications," International Journal of Agricultural and Biological Engineering, vol. 8, no. 2, pp. 72–79, 2015.
7. E. H. Mamdani, "Application of fuzzy algorithms for control of simple dynamic plant," Proceedings of the Institution of Electrical Engineers, vol. 121, no. 12, pp. 1585–1588, 1974.
8. E. H. Mamdani, "Application of fuzzy logic to approximate reasoning using linguistic synthesis," IEEE Transactions on Computers, vol. C-26, no. 12, pp. 1182–1191, 1977.
9. Z.-Y. Zhao, M. Tomizuka, and S. Isaka, "Fuzzy gain scheduling of PID controllers," IEEE Transactions on Systems, Man and Cybernetics, vol. 23, no. 5, pp. 1392–1398, 1993.
10. Sanchez E.N., Becerra H.M. and Velez C.M. (2007) "Combining fuzzy, PID and regulation control for an autonomous mini-helicopter", Information Sciences 177, pp. 1999–2022.
11. Precup R.E. and Preitl S. (2007) PI-Fuzzy controllers for integral plants to ensure robust stability, Information Sciences 177, pp. 4410–4429.
12. Juang Y.T., Chang Y.T. and Huang C.P. (2008) "Design of fuzzy PID controllers using modified triangular membership functions", Information Sciences 178, pp. 1325–1333.
13. Tran, H.K.; Son, H.H.; Duc, P.V.; Trang, T.T.; Nguyen, H.-N. Improved Genetic Algorithm Tuning Controller Design for Autonomous Hovercraft. Processes 2020, 8, 66. https://doi.org/10.3390/pr8010066
14. Tran, H.K.; Chiou, J.-S.; Dang, V.-H. New Fusion Algorithm-Reinforced Pilot Control for an Agricultural Tricopter UAV. Mathematics 2020, 8, 1499. https://doi.org/10.3390/math8091499
15. Passino K.M. and Yurkovich S. (1998) Fuzzy Control, Addison-Wesley, Reading, MA.
16. Eiben, A.E.; Smith, J.E. Introduction to Evolutionary Computing; Springer: Berlin, Germany, 2003.
17. Yang, Xin-She, Engineering Optimization: An Introduction with Metaheuristic Applications, John Wiley&Sons, Inc. University of Cambridge, 2010.
18. Kennedy J. and Eberhart R.C. (1995) "Particle swarm optimization", Proceedings IEEE International Conference on Neural Networks IV pp. 1942–1948.
19. Clerc M. (1999) "The swarm and the queen: Towards a deterministic and adaptive particle swarm optimization", Proceedings of the ICEC, Washington, DC, pp. 1951–1957.
20. Eberhart R.C. and Shi Y (2000) "Comparing inertia weights and constriction factors in particle swarm optimization", Proc. of the IEEE Congress on Evolutionary Computation, USA pp. 84–88.
21. Bergh F. and Engelbrecht A.P. (2004) "A Cooperative approach to particle swarm optimization," IEEE Transactions on Evolutionary Computation, vol. 8, no. 3, pp. 225-239.
22. Huang T and Mohan AS (2005) "A hybrid boundary condition for robust particle swarm optimization", IEEE Antennas and Wireless Propagation Letters 4, pp. 112–117.
23. Passino,K.M., "Biomimicry of bacterial foraging for distributed optimization and control," IEEE Control Syst. Mag. 22 (2002) 52–67.

24. Juang C.F. (2004) “A hybrid of genetic algorithm and particle swarm optimization for recurrent network design”, IEEE System, Man, and Cybernetics: B 34 (2), pp. 997–1006.
25. Li L.L., Wang L. and Liu L.L. (2006), “An effective hybrid PSOSA strategy for optimization and its application to parameter estimation”, Applied Mathematics and Computation 179, pp. 135–146.
26. Ali A.F. and Tawhid M.A. (2016) “A Hybrid PSO and DE Algorithm for Solving Engineering Optimization Problems”, Appl. Math. Inf. Sci. 10, No. 2, pp. 431-449.
27. Chen CL, Lin YL, Feng YC (2018) "Optimization of large-scale economic dispatch with valve-point effects using a modified hybrid PSO-DSM approach", Journal of Marine Science and Technology. DOI: 10.6119/JMST-015-1215-1
28. Hsu CI, Wu SPJ, Chiu CC (2019), "A Hybrid Swarm Intelligence Approach for Blog Success Prediction", International Journal of Computational Intelligence Systems, Vol. 12, Iss. 2, pp. 571 – 579
29. Liu XiaoLong, Li RongJun and YangPing, "A bacterial foraging global optimization algorithm based on the particle swarm optimization," 2010 IEEE International Conference on Intelligent Computing and Intelligent Systems, 2010, pp. 22-27, doi: 10.1109/ICICISYS.2010.5658828.
30. Gu, Q., Yin, K., Niu, B., Chen, H. (2012). RFID Networks Planning Using BF-PSO. In: Huang, DS., Ma, J., Jo, KH., Gromiha, M.M. (eds) Intelligent Computing Theories and Applications. ICIC 2012. Lecture Notes in Computer Science(), vol 7390. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-31576-3_24
31. Kora, P., Kalva, S.R. Hybrid Bacterial Foraging and Particle Swarm Optimization for detecting Bundle Branch Block. Springer Plus 4, 481 (2015). https://doi.org/10.1186/s40064-015-1240-z
32. W. M. Korani, H. T. Dorrah and H. M. Emara, "Bacterial foraging oriented by Particle Swarm Optimization strategy for PID tuning," 2009 IEEE International Symposium on Computational Intelligence in Robotics and Automation - (CIRA), 2009, pp. 445-450, doi: 10.1109/CIRA.2009.5423165.
33. S.M. Abd-Elazim, E.S. Ali, “A hybrid Particle Swarm Optimization and Bacterial Foraging for optimal Power System Stabilizers design,” International Journal of Electrical Power & Energy Systems, Volume 46, 2013, Pages 334-341, https://doi.org/10.1016/j.ijepes.2012.10.047.
34. H.E.A. Ibrahim, F.N. Hassan, Anas O. Shomer, Optimal PID control of a brushless DC motor using PSO and BF techniques, Ain Shams Engineering Journal, Volume 5, Issue 2, 2014, Pages 391-398, ISSN 2090-4479, https://doi.org/10.1016/j.asej.2013.09.013.
35. El-Wakeel, Amged & A.kamel, & Abdel-hamed, Alaa. (2015). A Hybrid BF-PSO Optimization Technique for Optimal Tuning of PID Controller of a Permanent Magnet Brushless DC Motor. Electric Power Components and Systems. 43. 309-319. 10.1080/15325008.2014.981320.
36. Amit Kumar and Sathans, (2020), Hybrid Bacterial Foraging Enhanced PSO Algorithm for Load Frequency Control of Hydro-Thermal Multi-Area Power System and Comparative analysis, International Journal of Computing and Digital Systems, Vol. 9, Iss. 1, ISSN (2210-142X), pp:129-138.
37. Kiran, H.U., Tiwari, S.K. (2021). Hybrid BF-PSO Algorithm for Automatic Voltage Regulator System. In: Gupta, D., Khanna, A., Bhattacharyya, S., Hassanien, A.E., Anand, S., Jaiswal, A. (eds) International Conference on Innovative Computing and Communications. Advances in Intelligent Systems and Computing, vol 1166. Springer, Singapore. https://doi.org/10.1007/978-981-15-5148-2_13
38. Skogestad, S. “Simple analytic rules for model reduction and PID controller tuning”, Journal of Process Control, Vol. 13, pp. 291-309, 2003.

39. Martins, F. G. "Tuning PID Controllers using the ITAE Criterion" International Journal Engineering Education, 21(5), pp. 867-873, 2005.
40. Tan, W.; Liu, J.; Chen, T.; Marquez, H.J. "Comparison of some well-known PID tuning formulas". Computation Chemical Engineering, pp. 1416–1423, 2006.
41. Tran, HK, Chiou, JS, Peng, ST. Design genetic algorithm optimization education software based fuzzy controller for a tricopter fly path planning. EURASIA Journal of Mathematics, Science and Technology Education 2016; 12(5): 1303–12.

Fusion Bacterial Foraging and Particle Swarm Optimization Algorithm For An Autonomous Hovercraft Control

Huu Khoa Tran[1], Pham Duc Lam[2], Tran Thanh Trang[3], and Hoang Nam Nguyen[4,*]

[1] Department of Information Management, Tamkang University, New Taipei City, Taiwan; email: hk5tran@mail.tku.edu.tw, https://orcid.org/0000-0001-5268-2355

[2] Faculty of Engineering and Technology, Nguyen Tat Thanh University, Vietnam. email: pdlam@ntt.edu.vn

[3] Faculty of Electrical and Electronics Engineering, Ho Chi Minh City University of Food Industry, Ho Chi Minh City, Vietnam; trangtt@hufi.edu.vn

[4] Modeling Evolutionary Algorithms Simulation and Artificial Intelligence, Faculty of Electrical and Electronics Engineering, Ton Duc Thang University, 19 Nguyen Huu Tho Street, Tan Phong Ward, District 7, Ho Chi Minh City, Vietnam;

* Correspondence: nguyenhoangnam@tdtu.edu.vn

Abstract: This article proposes a novel method for designing an effective autonomous Hovercraft vehicle controller. In the first step, we design a PD like Fuzzy controller, in which Proportional and Derivative (PD) parameters are used as the Fuzzy Scaling Factor of the controller design. Next, Bacterial Foraging (BF) and Particle Swarm Optimization (PSO) are combined to create a fusion algorithm to optimize the proportional and derivative gains of the PD like Fuzzy controller. The integral of time multiplied square error (ITSE) fitness function is applied to find the minima of the controller design error criterion. Numerical simulations are then implemented on a real autonomous Hovercraft motion model. Results indicate that the proposed controller design is more stable, reliable and accurate than existing controllers.

Keywords: Bacterial Foraging (BF) Algorithm, Particle Swarm Optimization (PSO), PD like Fuzzy, Hovercraft, Integral of Time Multiplied Square Error (ITSE).

1. Introduction

In recent years, many researchers have focused on using nonlinear control to build an autonomous Hovercraft [1–6] that can work well on various surfaces such as water, sand beach, ice and especially on muddy land. Most of these available controllers of the real hovercraft models are just based on the traditional PD controller and very few methods of optimizing the controller parameters are introduced for this system. The main aim of this paper is to propose a new way to optimize the proportional and derivative gains of the PD controller of the hovercraft.

Based on expert knowledge system, the Fuzzy Logic Controllers (FLC) have the advantage solution to the issues that could realize by human operators. Many works [7-14] has utilize this technique to solve various control system subjects and obtain considerably better results compare to others. When physical process has uncertain information and limited data, Fuzzy rule-based model is an appropriate choice. Therefore, we also use this approach in the hovercraft model control system. In this paper, we design a set of Fuzzy triangular membership functions and optimizes Scaling Factors (SF) as the inputs of Fuzzy controller. Although evolutionary algorithms [15, 16] such as Particle Swarm Optimization (PSO), Simulated Annealing (SA) and Genetic Algorithm (GA), can easily detect the local minima but it is difficult to detect the global optimization. Particle swarm optimization (PSO) [17-21] is a population-based heuristic algorithm that is invented based on observing animal swarm social behavior to detect accurate targets in multi-dimensional space. PSO uses particles (individuals), which are iteratively updated in each iteration, to perform searches. For finding the optimal solution, each particle decides its search direction based on its best previous location (cognitive part) and all other members' best locations (social part). In 2002, K. M. Passino [22], inspired from the Bacteria Foraging (BF) behavior of *Escherichia-coli* bacteria, firstly proposed a bionic algorithm that is very effective for distributed optimization and control design. Although, BF algorithm is very effective in local search but it is easily tapped or delayed in reaching global solution because of the random search directions during chemotaxis process. Due to eliminating the local optima trap and improving the optimal process, the modern engineering application researchers have introduced various PSO-based algorithms to solve the problems, for instance: GA and PSO [23], PSO and SA [24], hybrid PSO-Differential Evolution [25], Direct Search method and hybrid PSO [26], hybrid Swam Intelligence approach [27]. Also, combining PSO with bacterial foraging has emerged as one of the most efficient optimization methods, which can be applied successfully in many fields, including mathematical optimization [28], RFID network planning [29], medical care [30] and traditional PID control [31-36] in electric power systems. The above techniques can expand the optimal convergence pace by taking the advantages while eliminating the weakness of the solo algorithms.

Thus, we developed a fusion algorithm, called fusion Bacterial Foraging and Particle Swarm Optimization (named hereafter fBF-PSO), to increase the convergence speed in shorter operating generation and adjust the global search efficiency. We use PD control to improve hovercraft system stability. Additionally, a Fuzzy Inference System that takes advantage of PID conventional control, is utilized for reducing the steady state error. The fBF-PSO technique is also used to improve the performance of this PD like Fuzzy control by minimizing the ITSE fitness cost function [37-39]. The proposed controller can obtain the high-performance control of the Hovercraft in terms of higher accuracy, quick response, high stability and smoother maneuver. To the best of our knowledge, this is the first paper that combines the advantages of PSO and BF method in designing an effective and optimal PD controller for a real hovercraft model. In section 2, we describe the hovercraft configuration. In section 3, optimization algorithm and control strategy are introduced. In section 4, numerical simulation control of a real hovercraft model is presented. Finally, conclusions are discussed in section 5.

2. Autonomous Hovercraft configuration

The Air Cushion Vehicle, which is usually called as hovercraft, is composed of rotors and a cushion. Blower (rotor duct fan) is used to provide a large volume of air inside the air cushion, which is larger than the atmospheric pressure. Therefore, the hovercraft can float and is capable of travelling over ice, water, land, mud… [1-6, 13]. A tilt servo motor, which is located at the rear, is usually employed to steer the hovercraft. Although many modern technologies are utilized, there is still needs for a more advanced hovercraft maneuvering system with better performance. Especially, faster response and higher reliability are of great interest. Figure 1 shows a popular hovercraft model used in this research, which has a single tilt servo motor, a propeller settled along z-axis and a blower attached along y-axis.

The dynamic Hovercraft model utilized in this paper is referenced from [1, 6 and 13] with right hand coordinate systems. X-axis is used to position the lateral direction for controlling the sway motion and surge position. The Z-axis is in vertical direction and its positive direction is downwards. Whilst Y-axis is along its body, which is necessary for controlling sway motion or surge position. The hovercraft's kinematics can be described as equation (1):

$$\begin{cases} \dot{x} = p\cos\varphi - s\sin\varphi \\ \dot{y} = q\cos\varphi - s\sin\varphi \\ \dot{\varphi} = \omega \end{cases} \tag{1}$$

where $\omega \in R$ represents the angular velocity. $p,q,s \in R$ are defined as linear velocities in surge direction, sway direction and steering, respectively. From this above equation, we can derive the kinetic and potential energies of the hovercraft to compute L'agrange $L=T-V$. Then we can apply Euler-Lagrange formulation on equation (2):

$$\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\boldsymbol{q} = \begin{bmatrix} \boldsymbol{F} \\ \boldsymbol{T_q} \\ \boldsymbol{0} \end{bmatrix} \tag{2}$$

where $\boldsymbol{T_q} \in \boldsymbol{R}$ represents the torque in yaw and $\boldsymbol{F} \in \boldsymbol{R}$ is the control force in the surge direction. The torque control, which is perpendicular from the center of the hovercraft propeller, is a function of $\boldsymbol{F} \in \boldsymbol{R}$.

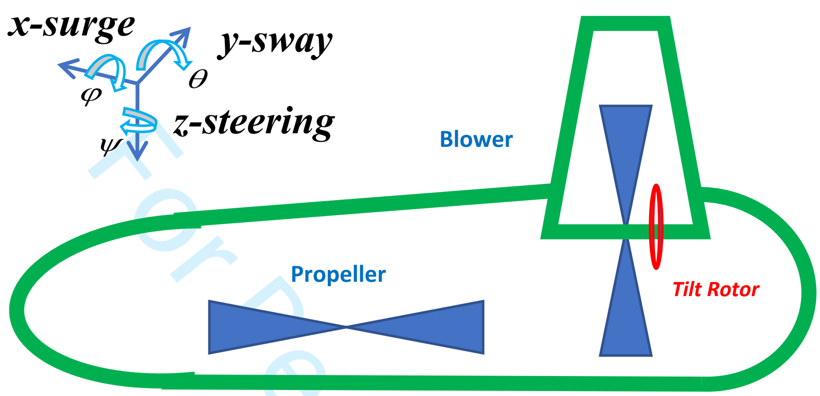


Figure 1. The Hovercraft prototype model.

3. Optimization Algorithms and Control Strategy

3.1.1. PD (Proportional and Derivative) *controller*

The PD control, which implemented in this article, is high efficiency despite owns a simplistic structure. Its design can reduce the settling time whereas improve the system stability. For the ith area, the PD controller combined gain, $K(s)$, is computed from the following equation (3):

$$\boldsymbol{G(s) = K_P + K_D.s} \tag{3}$$

Where, K_P and K_D are the conventional PD controller proportional and derivative gains, respectively.

3.1.2. Fuzzy Logic Controller (FLC)

The FLC dynamic behavior are based on a set of linguistic rules and originated from expert knowledge [7-9]. Designer needs to decide the input and out variables for building a suitable set of Fuzzy rules. In this article, the error *e(t)* and the error rate *de(t)/dt* are chosen as input variables, while the output is considered as c_i. Then, the relationship between these two inputs and one output variables are investigated. The error and error rate as well as the Fuzzy rules determine the online change of the system output c_i. We then need to fuzzify and defuzzify *e(t), de(t)/dt and* c_i

parameters. Fig. 2 illustrate the Fuzzy Inference System employed in this work. The Center of Area method defuzzification and Mamdani's MIN–MAX inference engine is utilized. Seven linguistic triangular membership functions for the two inputs and one output are assigned: positive big PB, positive medium PM, positive small PS, zero ZE, negative small NS, negative medium NM, and negative big NB. Table 1 explains the Fuzzy controller rules.

3.1.3 PD like Fuzzy controller

An intelligent control apply to autonomous hovercraft is the PD like Fuzzy controller. In that, FLC could be used as a classical PD controller, and overcome the disadvantages of the PD Controller. It is necessary to select the input and output variables and proper controller rules, as illustrated on equation (4)

$$\boldsymbol{u(t) = K_P.e(t) + K_D.de(t)} \quad (4)$$

Where K_P and K_D are proportional gains and differential gain which called as the scaling factors, *e(t)* is the error and *de(t)* is the change in error. Due to optimize the Scaling Factors as the inputs of PD like Fuzzy controller, the proposed optimization algorithms are presented on the next sub-sections.

Table 1. Rule base of Fuzzy Logic Controllers

Control Output CI (t)		***Input e(t)***						
		NB	**NM**	**NS**	**ZE**	**PS**	**PM**	**PB**
Input de(t)	**NB**	**ZE**	PS	PM	PM	**PB**	**PB**	**PB**
	NM	NS	**ZE**	PS	PM	PM	**PB**	**PB**
	NS	NM	NS	**ZE**	PS	PM	PM	**PB**
	ZE	NM	NM	NS	**ZE**	PS	PM	PM
	PS	**NB**	NM	NM	NS	**ZE**	PS	PM
	PM	**NB**	**NB**	NM	NM	NS	**ZE**	PS
	PB	**NB**	**NB**	**NB**	NM	NM	NS	**ZE**

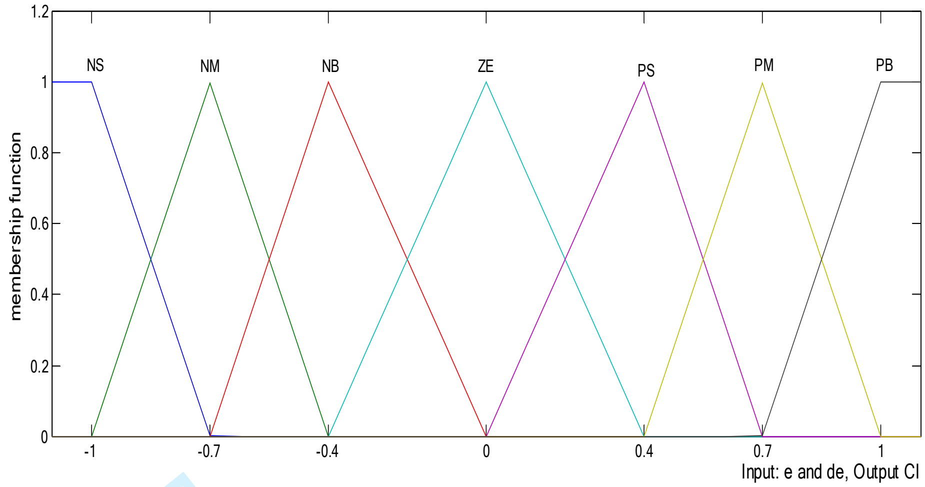


Figure 2. Fuzzy membership function.

3.2.1. PSO Algorithm

Recently, the PSO has emerged as the one of the most well-known and powerful tools for optimization [17-21]. We try to adapt and improve PSO algorithm for the hovercraft control. Specifically, PSO algorithm is combined with another effective optimization algorithm to optimally tune the PD controller gains. The PSO implementation process is illustrated by the flow chart in Fig. 3. The PSO mathematical equations (5) and equation (6) are briefly described as follows:

$$\boldsymbol{V(t+1) = \omega V(t) + c_1 r_1 (Pbest(t) - P(t)) + c_2 r_2 (Gbest(t) - P(t))} \quad (5)$$

$$\boldsymbol{P(t+1) = P(t) + V(t+1)} \quad (6)$$

Where V is particle velocity, P is the current position, Pbest is local best position while Gbest is global best position; ω is the inertia weighting factor, c_1 and c_2 as learning rates. The variables r_1 and r_2 are random distribution values ϵ [0 - 1].

3.2.2. BF optimization

The BF optimization derives from the searching foraging of the Escherichia **(E-coli)** bacteria [22] capabilities to survival in the natural changing environment. The fitness criteria of evolution process depend on their motile behavior to maintain the good foraging strategy as well as reshapes or even eliminate the poor strategy when they are on the path to finding the food source. The bacteria genes with the good foraging strategy are then proliferated in the evaluation chain and reproduced the better bacteria in next generations. The E-coli bacteria foraging progression to global searching capability is simplify described by four significant steps of Chemo-tactic (θ_k), Swarming, Reproduction (N_r) and Elimination–Dispersal. First, Chemo-tactic (θ_k) process illustrates the E-coli bacteria motion by two different methods; one can swim to fixed time while the other can tumble to alternates between two operation of the lifetime modes. Where, θ_k symbolizes the k^{th} bacterium, N_k is the size of the step taken in the random direction specified by the tumble and Δ is a length of the random direction unit vector, as shown in equation (7). Second, Swarming step: bacteria move to follow the nutrient gradient produced by the nutrient consuming group. Bacteria will release an attractant when they reach high nutrient areas and form concentric patterns of higher density bacteria. Third, Reproduction (N_r) step**:** **t**o maintain the constant population, the healthier bacteria split into two and the less healthy bacteria die. As shown in equation (8), Sr is the number of population members that have had enough nutrients to reproduce (split in two) without mutation. Finally, Elimination and Dispersal step: may occur when local significant increases in heat kill bacteria that are currently in nutrient-rich regions. Bacteria can be dispersed by an abrupt flow of water. Elimination and dispersal events may destroy chemotactic progress, but they may also promote chemotaxis, since they place bacteria near food sources. Soon after BF algorithm was invented, it has been widely used for optimization in numerous science and technology fields.

$$\boldsymbol{\theta_k} = \boldsymbol{\theta_k} + \boldsymbol{N_k} * \Delta_{\boldsymbol{k}} \tag{7}$$

$$\boldsymbol{S_r} = \boldsymbol{S/2} \tag{8}$$

3.2.3. Fusion Bacterial Foraging-Particle Swarm Optimization algorithm

Several studies [28-36] combined the ability of PSO to exchange social information with the ability of BF to eliminate and disperse old problems into new solutions**.** This combination was called as fBF-PSO, which has the combined strength of the two algorithms. To achieve optimal convergence, the proposed fBF-PSO algorithm is derived from the specified search directions of bacteria tumbling, which are oriented simultaneously by the individual and global best locations, and the identified ideas of PSO is updating position and velocity. This means that the proposed algorithm utilizes a PSO velocity calculation to precisely determine the updated chemotaxis drop direction within the BF algorithm. This allows tumble's unity-length random order to be adapted to the best global as well as the best individual position.

Several parameters must be evaluated, including iterations, inertia weight, learning rates, the number of bacteria, chemotactic steps~~, limit swimming~~ distance, reproduction steps, elimination-dispersal events, and probability that each bacterium is eliminated are referred from [16-19, 22, ~~28-36] and the own experimental controller designs. Hence, we combine fBF-PSO and the PD~~ like Fuzzy control for building the hovercraft controller. This algorithm is then implemented to find the SF as the best optimization parameters of the proposed controller. In the control diagram illustrated in Fig. 3, the tracking error e(t) and differential tracking error de(t) are employed as inputs of the Fuzzy Inference System. Two main parameters K_P (Proportional gain) and K_D (Derivative gain) of the PD like Fuzzy control are then optimally tuned with the Integral of Time Multiplied Square Error (ITSE) performance index [37-39]. After this, the results are compared qualitatively and quantitatively in terms of the ITSE minimum value: the settling time, the rising time, the overshoot and the tracking error, as demonstrated in equation (9):

$$\boldsymbol{ITSE} = \int_{\boldsymbol{0}}^{\infty} \boldsymbol{e^2(t)t}\,\mathbf{dt} \tag{9}$$

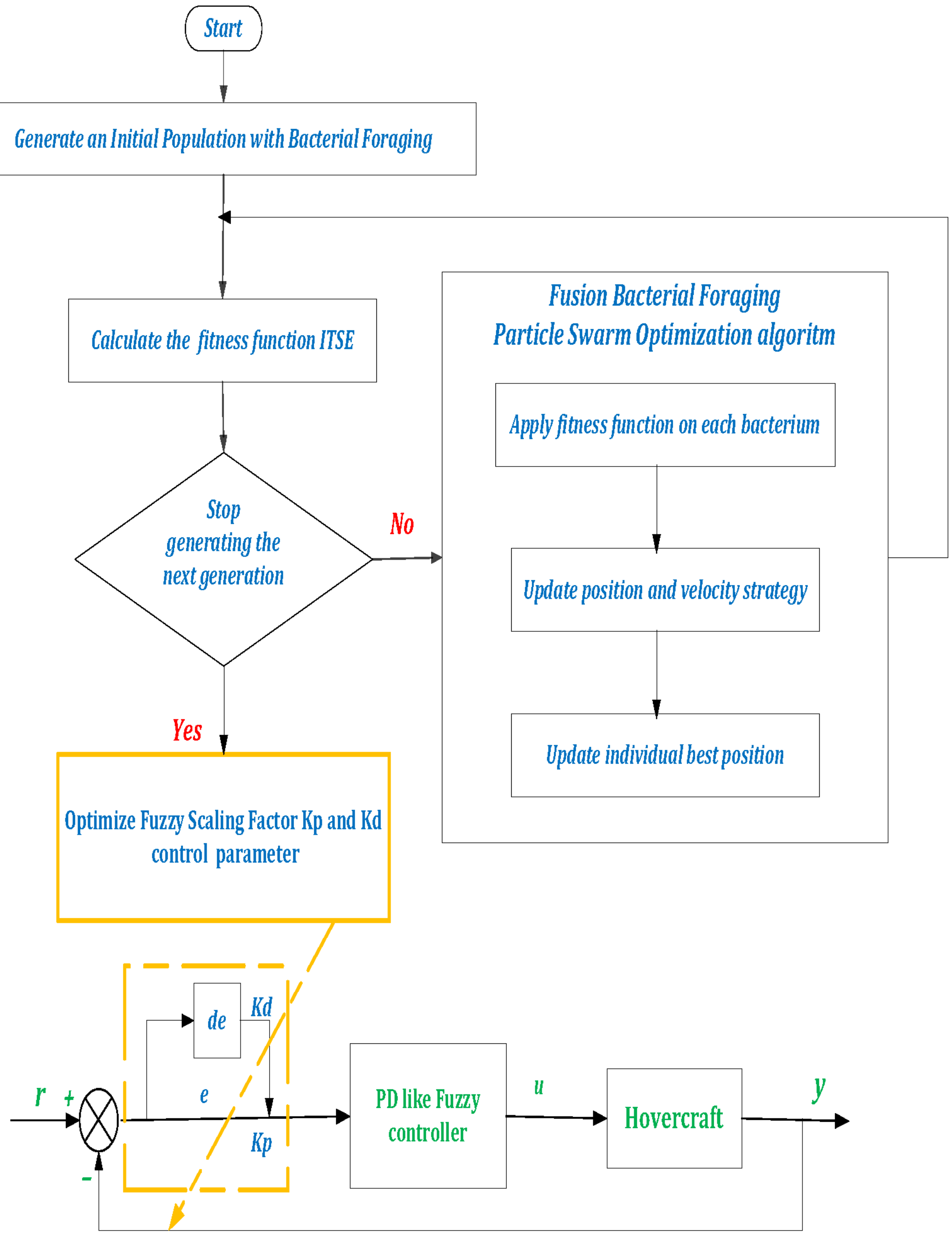


Figure 3. The fBF-PSO algorithm diagram.

4. Numerical Peformance

In this section, MATLAB/Simulink software is used to evaluate the performance of the proposed controller designs for autonomous Hovercraft mathematical models. These models were derived from the previous research [1, 6 and 13] with the mass of Hovercraft $m = 2.1\ kg$ and the inertia moment $I = 0.000257$. The simulation sampling time is 0.01 second. Numerical simulation results are operated in the stochastic optimization space. The proposed optimizations are simply set on 50 iterations. The PD gains of the proposed controllers are set in range ϵ [0, 100]. The significant optimal parameters of each optimization algorithm are chosen and displayed in the Table 2.

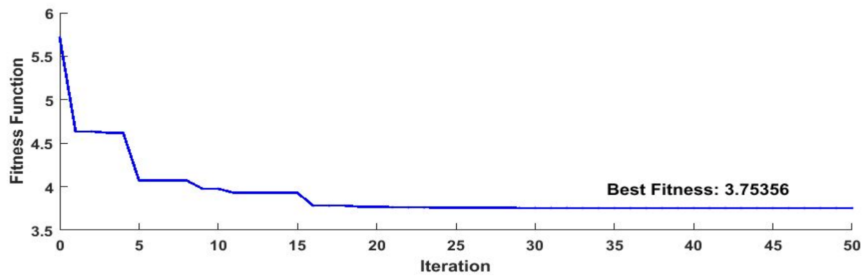
Table 2. Algorithms parameters setup

Parameters	PSO	BF	fBF-PSO
Dimension of search space	2	2	2
Generation or Iteration	50	50	50
Acceleration coefficient, $c1$	1.22	-	1.22
Acceleration coefficient, $c2$	1.22	-	1.22
Inertia weighting factor	0.92	-	0.92
Number of bacteria	-	50	50
Number of chemotactic steps	-	10	10
Limits the length of a swim	-	5	5
Number of reproduction steps	-	5	5
Number of elimination-dispersal events	-	3	3
Probability of each bacteria getting eliminated or dispersed	-	0.2	0.2

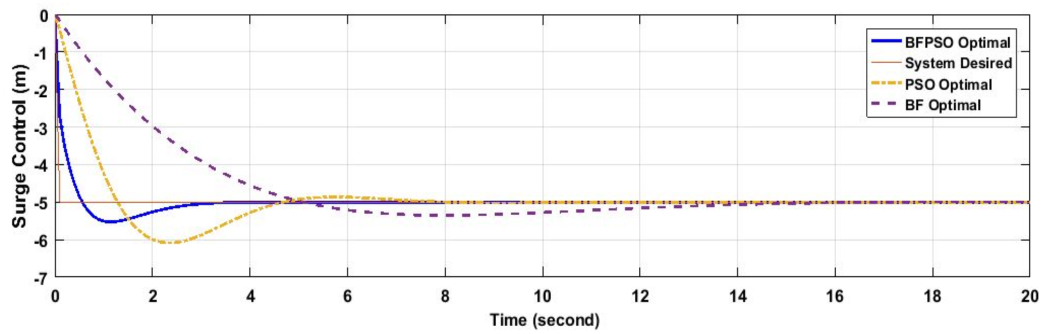
For the purpose of comparing the fBFPSO with the standard BF and conventional PSO methods in shorter generations of updating the fitness cost function, this paper investigate the performances of three channels of autonomous Hovercraft motions: Surge position x in Figure 4, Sway position y in Figure 5 and Yaw Steering angle in Figure 6, respectively. The numerical performances display that the fBFPSO methodology achieved the significant stability and dominance in performance response compare to the solo BF or the single PSO algorithm. All presented results proved that fBFPSO achieved the optimal control including: response time, less error, just after the third second at all, and were illustrated on the Table 3. Generally, all of the optimal controller designs are achieved the high-quality performance in terms of stability, high precision and high reliability.

Table 3. The fBF-PSO proposed method tuning gain results

	Surge position x (5 m)	***Sway position y (5 m)***	***Steering-Yaw angle (10 degree)***
Generation	*50*	*50*	*50*
K_P	*78.324*	*45.783*	*34.569*
K_D	*50.671*	*88.123*	*77.257*
Settling time (s)	*2.943*	*2.818*	*3.154*
Overshoot (%)	*10.12*	*1.05*	*0.93*
Elapsed time (s)	***21009.3397***	***29053.3585***	***25266.4916***

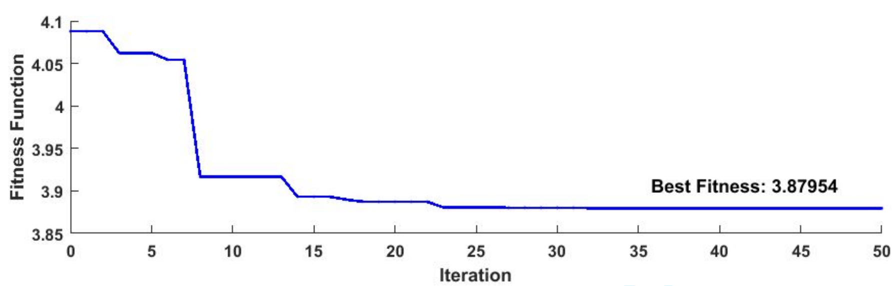


(a)

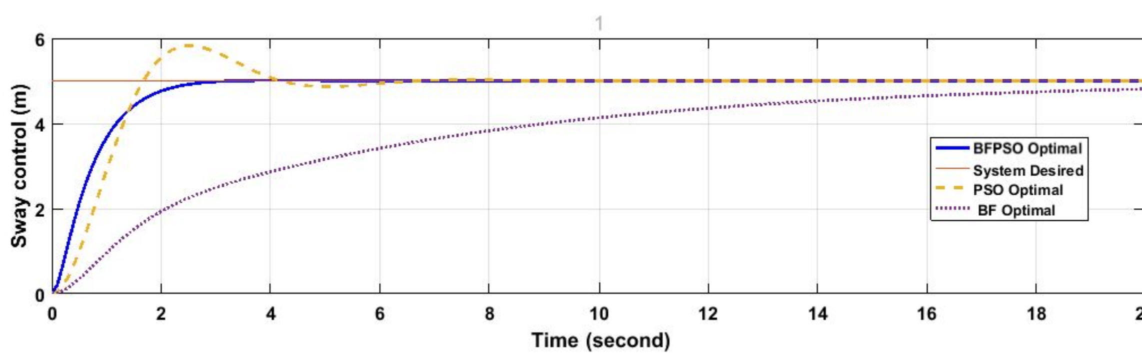


(b)

Figure 4. Hovercraft Surge control. (a) fBFPSO algorithm fitness value, (b) The proposed algorithm.

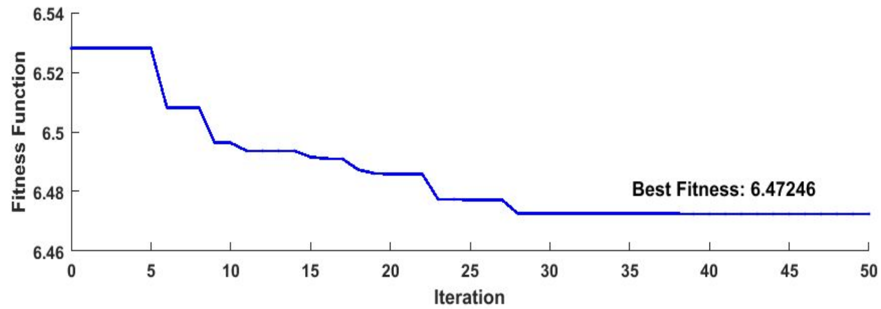


(a)

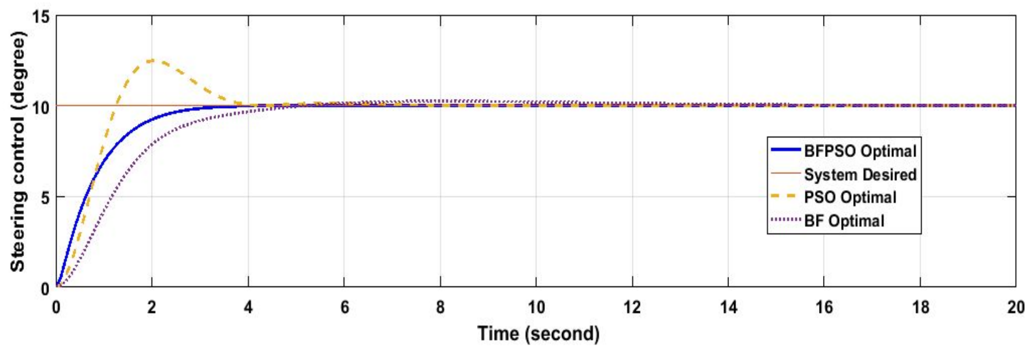


(b)

Figure 5. Hovercraft Sway control. (a) fBFPSO algorithm fitness value, (b) The proposed algorithm



(a)



(b)

Figure 6. Hovercraft Steering angle control. (a) fBFPSO algorithm fitness value, (b) The proposed algorithm.

5. Conclusions

The paper presents a successful design of three numerical optimal controllers to pilot an autonomous hovercraft. The fusion BF-PSO algorithm that employs both the ability to exchange social information of the PSO algorithm and the capability of finding new BF solutions via the elimination and dispersal step is proposed in order to find the optimal Fuzzy- like PD controller parameters for the hovercraft pilot models: surge, sway and steering control. The proposed controller obtains optimal pilot performance after the third second when compared with the other single optimal controller designs, PSO or BF. Furthermore, it provides fastest response, high reliability as well as stability than available controllers for the realistic hovercraft mathematical models. A path planning algorithm for Hovercraft model with obstacle avoidance and wind disturbances will be developed in future. In addition, the real hovercraft model with the proposed control technique will be implemented for testing and optimization.

References

1. D. Cabecinhas, P. Batista, P. Oliveira and C. Silvestre, "Hovercraft Control with Dynamic Parameters Identification," in IEEE Transactions on Control Systems Technology, vol. 26, no. 3, pp. 785-796, May 2018, doi: 10.1109/TCST.2017.2692733.
2. Hebertt Sira-Ramirez and Carlos Aguilar Ibanez, “On the control of the Hovercraft”. Dynamic and Control, Vol. 10, pp. 151-163, Kluwer Academic Publishers, 2000.

3. Silvano Balemi, Robert Bucher, Paola Guggiari, Ivan Furlan, Markus Kottmann and Jacques Chapuis, "Rapid control of prototyping of a Hovercraft", Conference: Proc. of MSy'02, Embedded Systems Conference, October, 2002.
4. Rashid, Aras, Kassim, Ibrahim, Jamali, "Dynamic Mathematical Modeling and Simulation Study of Small-scale Autonomous Hovercraft", International Journal of Advanced Science and Technology, Vol. 46, September, 2012.
5. Dictino Chaos, David Moreno-Salinas, Rocío Muñoz-Mansilla, and Joaquín Aranda, "Nonlinear Control for Trajectory Tracking of a Nonholonomic RC-Hovercraft with Discrete Inputs," Mathematical Problems in Engineering, vol. 2013, Article ID 589267, 16 pages, 2013.
6. Garcia D. I. and White W. N., "Control design of an unmanned hovercraft for agricultural applications," International Journal of Agricultural and Biological Engineering, vol. 8, no. 2, pp. 72–79, 2015.
7. E. H. Mamdani, "Application of fuzzy algorithms for control of simple dynamic plant," Proceedings of the Institution of Electrical Engineers, vol. 121, no. 12, pp. 1585–1588, 1974.
8. E. H. Mamdani, "Application of fuzzy logic to approximate reasoning using linguistic synthesis," IEEE Transactions on Computers, vol. C-26, no. 12, pp. 1182–1191, 1977.
9. Z.-Y. Zhao, M. Tomizuka, and S. Isaka, "Fuzzy gain scheduling of PID controllers," IEEE Transactions on Systems, Man and Cybernetics, vol. 23, no. 5, pp. 1392–1398, 1993.
10. Sanchez E.N., Becerra H.M. and Velez C.M. (2007) "Combining fuzzy, PID and regulation control for an autonomous mini-helicopter", Information Sciences 177, pp. 1999–2022.
11. Precup R.E. and Preitl S. (2007) PI-Fuzzy controllers for integral plants to ensure robust stability, Information Sciences 177, pp. 4410–4429.
12. Juang Y.T., Chang Y.T. and Huang C.P. (2008) "Design of fuzzy PID controllers using modified triangular membership functions", Information Sciences 178, pp. 1325–1333.
13. Tran, H.K.; Son, H.H.; Duc, P.V.; Trang, T.T.; Nguyen, H.-N. Improved Genetic Algorithm Tuning Controller Design for Autonomous Hovercraft. Processes 2020, 8, 66. https://doi.org/10.3390/pr8010066.
14. Passino K.M. and Yurkovich S. (1998) Fuzzy Control, Addison-Wesley, Reading, MA.
15. Eiben, A.E.; Smith, J.E. Introduction to Evolutionary Computing; Springer: Berlin, Germany, 2003.
16. Yang, Xin-She, Engineering Optimization: An Introduction with Metaheuristic Applications, John Wiley&Sons, Inc. University of Cambridge, 2010.
17. Kennedy J. and Eberhart R.C. (1995) "Particle swarm optimization", Proceedings IEEE International Conference on Neural Networks IV pp. 1942–1948.
18. Clerc M. (1999) "The swarm and the queen: Towards a deterministic and adaptive particle swarm optimization", Proceedings of the ICEC, Washington, DC, pp. 1951–1957.
19. Eberhart R.C. and Shi Y (2000) "Comparing inertia weights and constriction factors in particle swarm optimization", Proc. of the IEEE Congress on Evolutionary Computation, USA pp. 84–88.
20. Bergh F. and Engelbrecht A.P. (2004) "A Cooperative approach to particle swarm optimization," IEEE Transactions on Evolutionary Computation, vol. 8, no. 3, pp. 225-239.
21. Huang T and Mohan AS (2005) "A hybrid boundary condition for robust particle swarm optimization", IEEE Antennas and Wireless Propagation Letters 4, pp. 112–117.
22. Passino,K.M., "Biomimicry of bacterial foraging for distributed optimization and control," IEEE Control Syst. Mag. 22 (2002) 52–67.
23. Juang C.F. (2004) "A hybrid of genetic algorithm and particle swarm optimization for recurrent network design", IEEE System, Man, and Cybernetics: B 34 (2), pp. 997–1006.

24. Li L.L., Wang L. and Liu L.L. (2006), "An effective hybrid PSOSA strategy for optimization and its application to parameter estimation", Applied Mathematics and Computation 179, pp. 135–146.
25. Ali A.F. and Tawhid M.A. (2016) "A Hybrid PSO and DE Algorithm for Solving Engineering Optimization Problems", Appl. Math. Inf. Sci. 10, No. 2, pp. 431-449.
26. Chen CL, Lin YL, Feng YC (2018) "Optimization of large-scale economic dispatch with valve-point effects using a modified hybrid PSO-DSM approach", Journal of Marine Science and Technology. DOI: 10.6119/JMST-015-1215-1
27. Hsu CI, Wu SPJ, Chiu CC (2019), "A Hybrid Swarm Intelligence Approach for Blog Success Prediction", International Journal of Computational Intelligence Systems, Vol. 12, Iss. 2, pp. 571 – 579
28. Liu XiaoLong, Li RongJun and YangPing, "A bacterial foraging global optimization algorithm based on the particle swarm optimization," 2010 IEEE International Conference on Intelligent Computing and Intelligent Systems, 2010, pp. 22-27, doi: 10.1109/ICICISYS.2010.5658828.
29. Gu, Q., Yin, K., Niu, B., Chen, H. (2012). RFID Networks Planning Using BF-PSO. In: Huang, DS., Ma, J., Jo, KH., Gromiha, M.M. (eds) Intelligent Computing Theories and Applications. ICIC 2012. Lecture Notes in Computer Science(), vol 7390. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-31576-3_24
30. Kora, P., Kalva, S.R. Hybrid Bacterial Foraging and Particle Swarm Optimization for detecting Bundle Branch Block. Springer Plus 4, 481 (2015). https://doi.org/10.1186/s40064-015-1240-z
31. W. M. Korani, H. T. Dorrah and H. M. Emara, "Bacterial foraging oriented by Particle Swarm Optimization strategy for PID tuning," 2009 IEEE International Symposium on Computational Intelligence in Robotics and Automation - (CIRA), 2009, pp. 445-450, doi: 10.1109/CIRA.2009.5423165.
32. S.M. Abd-Elazim, E.S. Ali, "A hybrid Particle Swarm Optimization and Bacterial Foraging for optimal Power System Stabilizers design," International Journal of Electrical Power & Energy Systems, Volume 46, 2013, Pages 334-341, https://doi.org/10.1016/j.ijepes.2012.10.047.
33. H.E.A. Ibrahim, F.N. Hassan, Anas O. Shomer, Optimal PID control of a brushless DC motor using PSO and BF techniques, Ain Shams Engineering Journal, Volume 5, Issue 2, 2014, Pages 391-398, ISSN 2090-4479, https://doi.org/10.1016/j.asej.2013.09.013
34. El-Wakeel, Amged & A.kamel, & Abdel-hamed, Alaa. (2015). A Hybrid BF-PSO Optimization Technique for Optimal Tuning of PID Controller of a Permanent Magnet Brushless DC Motor. Electric Power Components and Systems. 43. 309-319. 10.1080/15325008.2014.981320.
35. Amit Kumar and Sathans, (2020), Hybrid Bacterial Foraging Enhanced PSO Algorithm for Load Frequency Control of Hydro-Thermal Multi-Area Power System and Comparative analysis, International Journal of Computing and Digital Systems, Vol. 9, Iss. 1, ISSN (2210-142X), pp:129-138.
36. Kiran, H.U., Tiwari, S.K. (2021). Hybrid BF-PSO Algorithm for Automatic Voltage Regulator System. In: Gupta, D., Khanna, A., Bhattacharyya, S., Hassanien, A.E., Anand, S., Jaiswal, A. (eds) International Conference on Innovative Computing and Communications. Advances in Intelligent Systems and Computing, vol 1166. Springer, Singapore. https://doi.org/10.1007/978-981-15-5148-2_13
37. Skogestad, S. "Simple analytic rules for model reduction and PID controller tuning", Journal of Process Control, Vol. 13, pp. 291-309, 2003.
38. Martins, F. G. "Tuning PID Controllers using the ITAE Criterion" International Journal Engineering Education, 21(5), pp. 867-873, 2005.

39. Tan, W.; Liu, J.; Chen, T.; Marquez, H.J. "Comparison of some well-known PID tuning formulas". Computation Chemical Engineering, pp. 1416–1423, 2006.