Hybrid Genetic Manta Ray Foraging Optimization and Its Application to Interval Type 2 Fuzzy Logic Control of An Inverted Pendulum System

Hybrid Genetic Manta Ray Foraging Optimization and Its Application to Interval Type 2 Fuzzy Logic Control of An Inverted Pendulum System

Ahmad Azwan bin Abdul Razak[1], Ahmad Nor Kasruddin bin Nasir[2], Nor Maniha Abdul Ghani[3], Shuhairie Mohammad, Mohd Falfazli Mat Jusof, Nurul Amira Mhd Rizal

Faculty of Electrical & Electronics Engineering Technology, Universiti Malaysia Pahang, 26600 Pekan Pahang, Malaysia.

ahmadazwan.ar@gmail.com[1], kasruddin@ump.edu.my[2], normaniha@ump.edu.my[3]

Abstract. This paper presents an improvised version of Manta-Ray Foraging Optimization (MRFO) by using components in Genetic Algorithm (GA). MRFO is a recent proposed algorithm which based on the behaviour of manta rays. The algorithm imitates three foraging strategies of this cartilaginous fish, which are chain foraging, cyclone foraging and somersault foraging to find foods. However, this optimization algorithm can be improved in its strategy which increases its accuracy. Thus, in this proposed improvement, mutation and crossover strategy from GA were adopted into MRFO. Crossover operation is a convergence action which is purposely to pull the agents towards an optimum point. At the meanwhile, mutation operation is a divergence action which purposely to spread out the agents throughout wider feasible region. Later, the algorithms were performed on several benchmark functions and statically tested by using Wilcoxon signed-rank test to know their performances. To test the algorithm with a real application, the algorithms were applied to an interval type 2 fuzzy logic controller (IT2FLC) of an inverted pendulum system. Result of the test on benchmark functions shows that GMRFO outperformed MRFO and GA and it shows that it provides a better parameter of the control system for a better response.

1. Introduction

Manta Ray Foraging Optimization (MRFO) is a recently population-based optimization algorithm [1]. It based on the strategy of a species in cartilaginous fishes called Manta Ray to find foods. Manta rays are among the largest marine creatures [2]. They have a large flat body from top to bottom with a pair of pectoral fins and swims as birds that freely fly. They also have a cephalic lobe that extend in front of their giant terminal mouth. Manta rays feed on plankton of microscopic organisms from the water. By using their cephalic lobe, their preys were funnelled into their mouths. Later, they filter the prey from the water by using their gill. There are two distinct species, one is reef manta ray, second is giant manta ray. First species can be found mainly in Indian ocean, western and south Pacific while second species can be found in tropical, subtropical and warm temperate oceans. A matured manta ray eats up to 5kg plankton in a single day [2]. On the far side, the plankton, however are not evenly dispersed or grouping at a certain area but formed according to the flow of the tides or changes of the seasons. Fascinatingly, manta rays always superb in finding those plankton in large quantity using their unique behaviour of foraging strategy. The strategy includes chain foraging, cyclone foraging and somersault foraging. Chain foraging is an action of a group of manta rays which are lining up, one behind another [1]. They will scoop the plankton in their ways, in which the plankton missed by previous manta ray will be scooped by one who is following behind. Cyclone foraging is a gathering of manta rays when they found a high concentration of plankton [1]. In this strategy, their tails link up with heads in spiral to produce a spiralling tide to moves plankton towards the water surfaces. In this strategy, while move in spiral, they open their mouth to funnel and eat the plankton. Apart from that, somersault foraging is a behaviour in which the manta ray will do a sequence of backwards somersault, circling around to draw the plankton that found towards them [1]. This is a random, frequent, local and cyclical movement which aid manta rays in food intake.

Genetic algorithm (GA) on the other hand is one of the well-known optimizers. It is a derivative-free and stochastic optimization methods based on the concept of evolution. To find optimum solution, GA adopts strategies inspired by evolutionary biology like mutation, crossover, selection and inheritance [3]. From the theory, evolution begins from a population of randomly created individuals and take places in generations. For each generation, the fitness of every individuals was measured. Based on their fitness, a number of individuals were selected and modified by mutation and crossover to create a new population. Mutation is an operation to modify individual's genetic information. Mutation is important to create a diverged population. It is designed to sometimes break one or more individuals out of local optima which lead to global optima [3]. In parallel, crossover take place when a pair of chromosomes break and then reconnect but to the distinct end piece. Crossover is significant to pull the individuals towards optimum point which lead to accuracy of the solution [4].

In this paper, a new variant called GMRFO is proposed. Mutation operation are interspersed into MRFO to improve the dispersion of agents in the feasible region and crossover to improves convergence rate of the individuals toward the optimum point. Thus, exploration and exploitation of MFRO will be well improved. GMRFO will used to optimize an IT2FLC of an Inverted Pendulum System (IPS) for controlling the angle of the pendulum and its cart position. The paper is organized as follows. Section 2 unveils the IPS and its parameters. Section 3 unveils the MRFO, GA and GMRFO. Section 4 describes two separated result section: 1) The experimental setup for benchmark test and 2) Performance analysis on IT2FLC. Section 5 discussed the results and Section VI concludes the proposed algorithm.

2. Inverted Pendulum System

Inverted pendulum system (IPS) is a classic control challenge and widely used in control study. It is a high non-linearity and lack of stability and make it adequate to examine new prototype controllers [5]. IPS composes of an inverted pole with specific mass hinged at certain angle from vertical axis on a cart with mass and free to move horizontally. The pole hinged on the cart moves linearly while the cart in movement and simultaneously another end of the pole rotates 360 degrees around the hinged pole. During static, the pole is pointing downward. IPS is a single-input multi-output system and usually a voltage is considered as the input, whereas position of the cart and the angle of pole are considered as the output of the system. Ordinarily, controller used to maintain the pole at upwards position while the cart is moving or remaining static. Several real-world applications like two-wheel chairs and two-wheel Segway transporter are the example of the application of IPS. IPS was managed to be controlled using PD-controller by proposing different proportional and derivatives gains. A root locus also drew to verify the stability. Nevertheless, there are limitation in this technique which are the vibration and insufficient current.

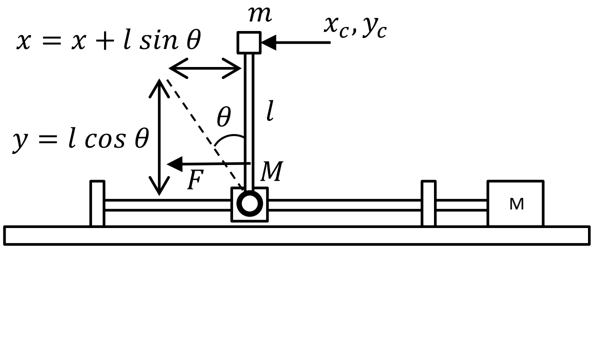

Figure 1. Inverted Pendulum System.

Table 1. Parameters For IPS.

Parameter	Values
Mass of cart, $\boldsymbol{M_{pen}}$	0.1kg
Mass of pendulum, $\boldsymbol{m_{pen}}$	0.05kg
Friction or cart, $\boldsymbol{f_{cart}}$	$0.1Nm^{-1}s^{-1}$
Length of pendulum, $\boldsymbol{l}$	0.3m
Inertia of the pendulum, $\boldsymbol{I}$	0.006 $kg\ m^2$
Motor torque constant, $\boldsymbol{K_m}$	4.9 $Ncm\ A^{-1}$
Motor back $\boldsymbol{emf}$ constant, $\boldsymbol{K_b}$	0.0507 $V\ rad^{-1}s^{-1}$
Motor armature resistant, $\boldsymbol{R}$	0.3Ω

Based on the Fig. 1 and Table 1, reflecting both mechanical and electrical components of IPS, derivation for equation of motion based on Newton 2nd Law can be represented as follow.

$$(M+m)\ddot{x} + ml\theta + F_r\dot{x} = F_v V \tag{1}$$

where,

$$F_r = b + \left(\frac{2\pi}{r}\right)^2 \left(\frac{K_m K_b}{R}\right) \tag{2}$$

$$F_v = \left(\frac{2\pi}{r}\right)\left(\frac{K_m}{R}\right) \tag{3}$$

$$m\ddot{x} + ml\ddot{\theta} = mg\theta \tag{4}$$

where, b is friction coefficient of the cart, r is transfer distance per one rotation of the ball screw, K_m is motor torque, K_b is back emf constant and R is motor armature resistance. The equation of motion then converted into state space equation.

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{x} \\ \ddot{x} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{(M+m)g}{Ml} & 0 & 0 & \frac{F_r}{Ml} \\ 0 & 0 & 0 & 1 \\ -\frac{mg}{M} & 0 & 0 & -\frac{F_r}{M} \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ x \\ \dot{x} \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{F_v u}{Ml} \\ 0 \\ \frac{F_v u}{M} \end{bmatrix} \tag{5}$$

$$[y] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ x \\ \dot{x} \end{bmatrix} \tag{6}$$

In Equation 5, u portrays as input, which is voltage into the system. At the meanwhile, in Equation 6, two outputs were gauged which is angle of the pole, θ and cart position, x.

3. Genetic-Manta Ray Foraging Optimization (GMRFO)

3.1. Manta Ray Foraging Optimization (MRFO)

MRFO is stimulated by three foraging behaviours of manta rays which are chain foraging, cyclone foraging and somersault foraging. This section explained the mathematical expression for these behaviours [1]. Denote that in this algorithm, the best solution is assumed as the plankton with high concentration manta rays want to approach and eat.

3.1.1. Chain Foraging. Chain foraging is the first behavior of manta ray in hunting for foods. They link in a line, where the manta in front scoop the plankton on their ways while the missed plankton will be scoop by the manta ray who follows behind. Equation (7) and (8) explained the mathematical expression for these behaviors [1].

$$x_i(k+1) = \begin{cases} x_i(k) + r_1(x_{best} - x_i(k) + \alpha(x_{best} - x_i(k)), i = 1 \\ x_i(k) + r_1(x_{i-1}(k) - x_i(k) + \alpha(x_{best} - x_i(k)), i = 2, 3, ..., n \end{cases} \tag{7}$$

where,

$$\alpha = 2r\sqrt{|\log(r_1)|} \tag{8}$$

where $x_i\ (k+1)$ is new position of individuals, $x_i\ (k)$ is the current position of i^{th} individuals at time k, r_1 is a random number within [0,1], α is a weight coefficient, x_{best} is the plankton with high concentration.

3.1.2. Cyclone Foraging. When the manta rays found a high concentration of plankton, they will move towards the food in spiral path and in chain simultaneously. This spiral model is mathematical expressed as follows. For initial state, manta rays move in random position using equation:

$$x_i(k+1) = \begin{cases} x_{rand} + r_2\left(x_{rand} - x_i(k) + \beta\left(x_{rand} - x_i(k)\right)\right), i = 1 \\ x_{rand} + r_2\left(x_{i-1}(k) - x_i(k) + \beta\left(x_{rand} - x_i(k)\right)\right), i = 2, 3, ..., n \end{cases} \tag{9}$$

At latter phase, manta rays will move towards the best individuals using equation [1]:

$$x_i(k+1) = \begin{cases} x_{best} + r_2\left(x_{rand} - x_i(k) + \beta\left(x_{best} - x_i(k)\right)\right), i = 1 \\ x_{best} + r_2\left(x_{i-1}(k) - x_i(k) + \beta\left(x_{best} - x_i(k)\right)\right), i = 2, 3, ..., n \end{cases} \tag{10}$$

where

$$\beta = 2e^{r_2 \frac{k_{\max}+1}{k_{\max}}} \sin(2\pi r_2) \tag{11}$$

where β is the weight coefficient, k_{max} is the maximum iterations and r_2 is the random number between [0,1].

3.1.3. Somersault Foraging. In this strategy, location of food is seen as a pivot. Manta rays will swim towards and backwards around the pivot and somersault to a new location. Thus, the manta ray will update their new location around x_{best} so far. The mathematical expression of somersault is followed [1],

$$x_i(k+1) = x_i(k) + S\left(r_3 x_{best} - r_4 x_i(k)\right) \tag{12}$$

where r_3 and r_4 are random number between [0,1] and S is the somersault range of manta rays.

3.2. Genetic Algorithm

GA was developed to simulate biological process inspired by theory of evolution [3,4,5]. The strategy is adopting the survivals of the fittest. Later, the fittest individual will tend reproduce and improvised by each generation. However, some of the inferior are allowed to survive by a little chance to reproduce. GA uses mutation and crossover to reproduce offspring. Mutation a genetic operator used to maintain genetic diversity from one generation of population of GA chromosomes to the next [3]. In GA, mutation operator alters the positions randomly between upper and lower bound. In the meanwhile, crossover is a genetic operator to combine genetic information of two individuals to generate new offspring. In GA, two individuals are selected stochastically to produce two new offspring. Mutation is an operation to diverse the individuals within feasible region and proved to be efficient to make sure all region was explored. Crossover is significant to converge the solution towards accuracy [3]. Synergy of these operators are the factor that make GA is well-known as reliable optimization algorithm to solve many applications.

3.3. Genetic-Manta Ray Foraging Optimization (GMRFO)

Genetic-Manta Ray Foraging Optimization (GMRFO) is an extended version of MRFO. GMRFO is modified by adding GA components, which are mutation and crossover operations. GMRFO requires no setup parameters as all of the constants used in GMRFO originally from GA like mutation and crossover rate were set adapted to the iteration number. Then, this proposed algorithm was run based on number of function evaluation (NFE) of 10000 times. This is important to provide a fair comparison against MRFO and GA.

```
Step 0: Preparation
Select a number of manta rays in a population, $nPop,$ maximum number of iterations, $k_{max}$ and function evaluation (NFE).

Step 1: Initialization
Initialize the location points of manta rays in random position $x_i(k),$ between feasible region.
    $x_i(k) = x_l + rand(x_u - x_l)$
where $x_l$ is lower bound and $x_u$ is upper bound.

Step 2: Evaluate Fitness.
Evaluate fitness, $f_i$ of manta rays for each location and acquire best solution so far as $x_{best}$.
    $f_i = f(x_i(k))$

Step 3: Apply Foraging Model Equation.
Choose a random number, $r_1$. If $r_1$ smaller then $k/k_{max}$, then do cyclone foraging. Otherwise, do chain foraging.

    a) Cyclone foraging
       Choose a random number, $r$. If $r < k/k_{max}$, then move individuals by using Equation (9).
       Else if $r > k/k_{max}$, then use Equation (10).
    b) Chain foraging
       Move individuals by using Equation (7).

Evaluate $f_i = f(x_i(k))$. Choose a new best manta ray, $x_{best}$.

If $f\big(x_i(k+1)\big) < f(x_{best})$, then, $x_{best} = x_i(k+1)$

Then apply somersault foraging using Equation (12).
Apply Mutation.
Apply Crossover.

Evaluate $f_i = f(x_i(k))$. Choose a new best manta ray, $x_{best}$.

Step 4: Checking Termination Criterion
Check if termination criterion, $k_{max}$ is met, then stop. Otherwise, return to Step 2.
```

Figure 2. Pseudocode of the proposed GMRFO.`

4. Experimental Setup

4.1. Benchmark Function

To test the performance of accuracy of the proposed algorithm, four different benchmark functions were used. They are function F1, F2, F3, and F4 which are Shifted and Rotated Weierstrass Function, Shifted and Rotated Katsuura Function, Shifted and Rotated Happy-Cat Function and Hybrid Function 4 respectively [7]. F1, F2, F3, are multimodal, non-separable problems. In the meanwhile, F4 is a hybrid of several functions. All of these functions are defined in the Table 2. To test its performance, these functions were defined as 10 dimensions (D) with search ranges between [-100,100]. Each function was run for 51 times by 10,000 times function of evaluations [7].

Table 2. Benchmark function used.

Function Number	Mathematical Expression
F1	$f_1(x) = \sum_{i=1}^{D}\left(\sum_{k=0}^{k_{max}}\left[a^k \cos\left[2\pi b^k \left(x_i + 0.5\right)\right]\right] - D\sum_{k=0}^{k_{max}}\left[a^k \cos\left(2\pi b^k .0.5\right)\right]\right)$
F2	$f_2(x) = \left\lvert\sum_{i=1}^{D} x_i^2 - D\right\rvert^{1/4} + \frac{\left(0.5\sum_{i=1}^{D} x_i^2 + \sum_{i=1}^{D} x_i\right)}{D} + 0.5$
F3	$f_3(x) = f_4\left(f_5\left(x_1, x_2\right)\right) + f_4\left(f_5(x_2, x_3)\right) + ... + f_4\left(f_5\left(x_{D-1}, x_D\right)\right) + f_4\left(f_5\left(x_D, x_1\right)\right)$ Where f_4 and f_5 are: $f_4(x) = \sum_{i=1}^{D}\left(x_i^2 - 10\cos(2\pi x_i) + 10\right)$ $f_5(x) = \sum_{i=1}^{D-1}\left(100\left(x_i^2 - x_{i+1}\right)^2 + \left(x_i - 1\right)^2\right)$
F4	$N = 4$ $p = [0.2, 0.2, 0.3, 0.3]$ g_1: HG-Bat Function g_2: Discus Function g_3: Expanded Griewank's plus Rosenbrock's Function g_4: Rastrigin's Function

4.2. Interval Type 2 Fuzzy Logic Control (IT2FLC) of The Inverted Pendulum System

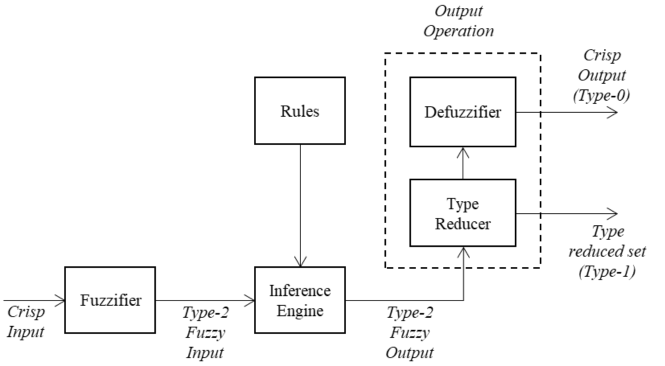

Figure 3. Structure of IT2FLC.

Figure 3 shows the structure of IT2FLC. Type-2 fuzzy sets membership function are fuzzy and consist footprint of uncertainties (FOU) that examines and represent the uncertainties, non-linearities and syntactical related to the input and output of the FLC. Denotes that, FOU is the area between lower and upper membership function. FOU allows each input to obtain a pair of membership grade: Upper Membership Function (UMF) and Lower MF (LMF). In this study, two closed-form representations of IT2FLC were used. They are based on approximating a method called Coupland and John's Geometric Centroid (GC) method. GC method is defuzzification that provides a good approximation to the type-reduced output. GC by-passes type-reduced centroid by directly finding the x-coordinate of the geometric centroid of the FOU of the output of the IT2FLS. Other than that, GC performed to find upper and lower bound independently and concurrently, while Type-1 operation can be utilized at the background.

4.2.1 Computation of IT2FLC [12]

To explain computation, consider a two-input single-output IT2FLC. In this study, there are possible $M = N^2$ rules and its k^{th} rule (denoted as R^{th}) has following form : R^k: $If\ x_1$ is $\tilde{F}_1^i$ and $x_2\ is\ \tilde{F}_2^j$, then y is $\tilde{G}^{ij}$; $(i,j = 1,2,3,\dots,N)$, where $k = (i-1)N + j, k \in [1, M]$. The k^{th} rule also present as the ij^{th} rule and can be presented as the centroid of the a consequent. The consequences also, $\tilde{G}^{ij}$ can be intervals or Type-1 fuzzy sets.

1. Set firing level of R^k by Equation (13). Denotes that, $\underline{f}^{ij}$ and $\bar{f}^{ij}$ are the firing level degrees for both lower and upper, $\underline{\mu}_{\tilde{F}_1^i}$ and $\bar{\mu}_{\tilde{F}_2^j}$ are the lower and upper membership grades of $\tilde{F}^i(x)$.

$$F^{ij} = \left[\underline{\mu}_{\tilde{F}_1^i}(x_1) * \underline{\mu}_{\tilde{F}_2^j}(x_2), \overline{\mu}_{\tilde{F}_1^i}(x_1) * \overline{\mu}_{\tilde{F}_2^j}(x_2) \right] \equiv \left[\underline{f}^{ij}, \overline{f}^{ij} \right] \tag{13}$$

2. Calculate the overall output IT2 FS by aggregation the implied set from the fired rules. Denotes that, ⊔ is the aggregation operation, * is the utilized t-norm operator, usually minimum or product operator. The mathematical expression stated in Equation (14).

$$\mu_{\tilde{B}}(y) = \coprod_{k=1}^{M} \mu_{\tilde{B}^k}(y) = \left[\coprod_{k=1}^{M} \left(\underline{f}^k * \mu_{G^k}(y) \right), \coprod_{k=1}^{M} \left(\overline{f}^k * \mu_{G^k}(y) \right) \right] \tag{14}$$

3. Find the centre of area between lower and upper bound, CoA which represent the crisp output of IT2FLS using Equation (15).

$$y_{crisp} = CoA(\tilde{B}) = \frac{\int_{-\infty}^{\infty} y.dA}{\int_{-\infty}^{\infty} dA} = \frac{\int_{-\infty}^{\infty} y.\left(\overline{\mu}_{\tilde{B}}(y) - \underline{\mu}_{\tilde{B}}(y) \right).dy}{\int_{-\infty}^{\infty} \left(\overline{\mu}_{\tilde{B}}(y) - \underline{\mu}_{\tilde{B}}(y) \right).dy} = \frac{\sum_{i=1}^{n} y_i \mu_{\tilde{B}}(y_i)}{\sum_{i=1}^{n} \mu_{\tilde{B}}(y_i)} \tag{15}$$

Figure 4 shows the close-loop block diagram of the IT2FLC of the inverted pendulum system (IPS). The IT2FLC has 9 rules, 2 sigma and 2 footprint of uncertainties (FOU). To control both angle and position, two sets of IT2FLC were include in the test. Therefore, there are total of 26 parameters that optimized in this study [8,9,10]. In this study, there are two output of interest which are angle and position. Emphasized that, output response will be compared to input. The different will be fitness value that applied to MRFO, GA and GMRFO. Both of them will search the correct parameters of IT2FLCs in order to optimize the error. The mathematical expression of the cost function of the MRFO and GMRFO is express in Equation (16)-(18).

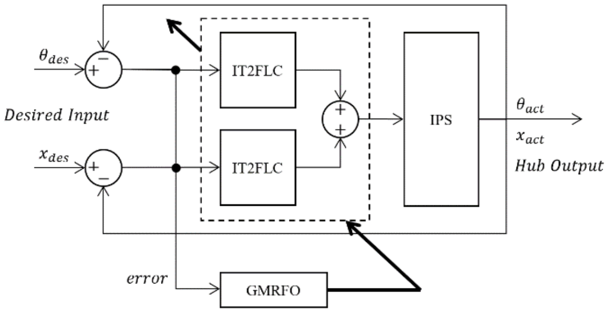

Figure 4. Close-loop IT2FLC of the Inverted Pendulum System.

$$J_{\theta} = \frac{1}{N}\sum_{i=1}^{N}\left(\theta_{desired\,(i)} - \theta_{output}\right) \tag{16}$$

$$J_{x} = \frac{1}{N}\sum_{i=1}^{N}\left(x_{desired\,(i)} - x_{output}\right) \tag{17}$$

$$J_{total} = J_{x} + J_{\theta} \tag{18}$$

5. Result and Discussion

5.1. Benchmark Function Analysis

In this section, performance of MRFO, GA and GMRFO in solving benchmark functions are compared. The convergence curves of for all algorithms were plotted in Figure (5)-(8). From the figures, all algorithms have almost same convergence speed at the initial state of searching. However, MRFO and GA stuck in local optima and stop to converge which lead to produce low accuracy solution. In contrast, GMRFO converge further and produce more accurate solution as the iteration increases. As conclusion, GMRFO can produce better solutions in term of accuracy for these problems.

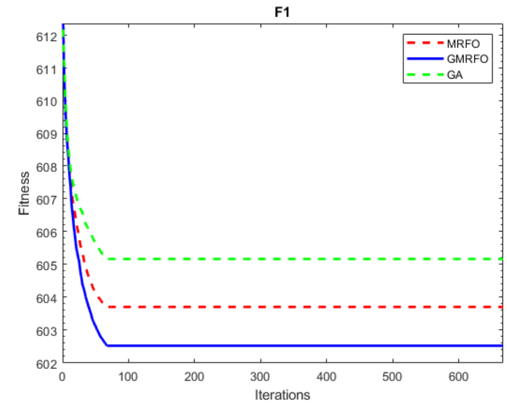

Figure 5. Convergence Curve for F1.

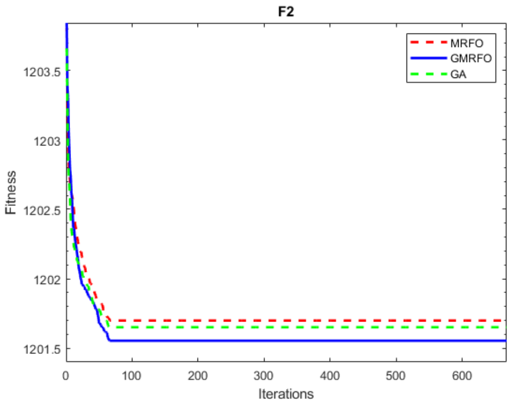

Figure 6. Convergence Curve for F2.

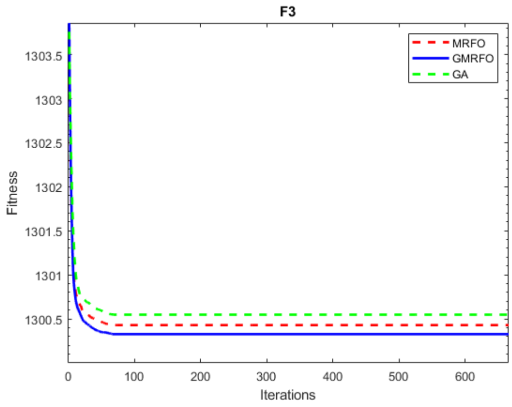

Figure 7. Convergence Curve for F3.

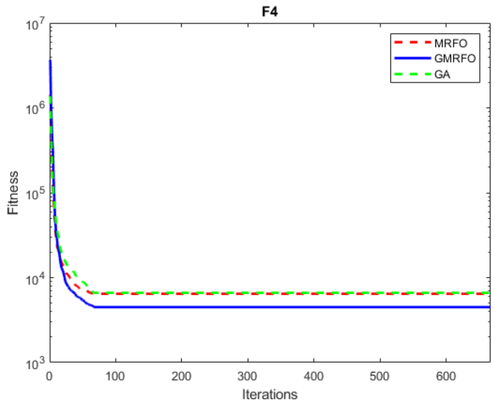

Figure 8. Convergence Curve for F4.

The numerical result of accuracy performance of the MRFO, GA and GMRFO is described in Table 3. The tables show the average value of the accuracy based on 51 times run. From the table, GMRFO obviously outperforms MRFO and GA.

The result then tested using Friedman and Wilcoxon Sign-rank test [11,13]. The Friedman test was used to rank all performance of the algorithms. The best and the worst are shown by the lowest value and vice-versa. At the meanwhile, Wilcoxon test was used to determine the significant improvement of the algorithms. The confidence level is set with 5% of *p*-value, in which if the test give value below than the confidence value, the result is considered as significantly improved. All the statistical results, mean rank from the Friedman test, and *p*-values from the Wilcoxon Signed Rank test are reported in Table 3.

Table 3. Statistical results, Friedman test, and Wilcoxon Signed Rank test.

	Statistical Result			Friedman Test			Wilcoxon Signed Rank Test			
No	Mean			Mean Rank			MRFO vs GMRFO		GA vs MRFO	
	MRFO	GA	GMRFO	MRFO	GA	GMRFO	Sign Rank	*p*	Sign Rank	*p*
1	6.0363E+02	6.0516E+02	**6.0229E+02**	1.84	2.65	**1.51**	904	0.024	1256	0.000
2	1.2017E+03	1.3217E+03	**1.2017E+03**	2.02	2.96	**1.02**	1326	0.020	1326	0.001
3	1.3004E+03	1.3005E+03	**1.3004E+03**	1.96	2.57	**1.47**	1074	0.000	1257	0.000
4	6.3779E+03	6.6277E+03	**3.7687E+03**	2.22	2.10	**1.69**	1020	0.001	973	0.004

Table 3 shows the solution provided by GMRFO has higher accuracy compared to MRFO and GA all test case. From the table, GMRFO also ranked first (lowest value among others) for all the benchmark function by Friedman test with significant improvement which were proved by Wilcoxon test. These statistical analyses were tally with the Figures 5-8.

5.2. Performance Analysis of tuned-IT2FLC on Inverted Pendulum System

This section present results for the application of MRFO, GA and GMRFO for their application on IPS. In this test, IPS were simulated to be at 10 cm away from its original position with the pendulum swung-up and stay at 0-radian. The MRFO, GA and the proposed GMRFO were applied to optimized the IT2FLC of 26 parameters including 18 rules, 4 sigma and 4 FOU. All of these constants were found in the feasible region between [-1,1]. This range was taken due to longer period of time is required if the feasible region is wider. All algorithms searched for the optimized values for those constants. This application was run for only three times due to it requires long period to finish a run.

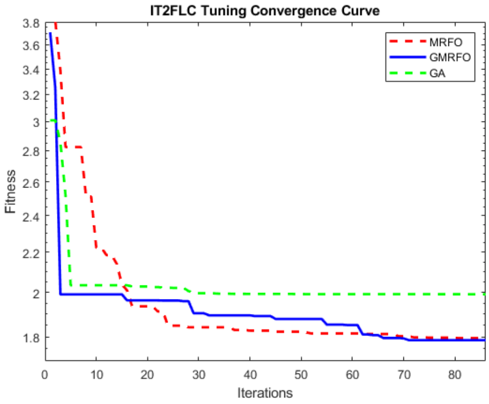

Figure 9. Convergence Curve for IT2FLC Tuning.

Figure 9 shows the convergence curve produced by all three algorithms. From the convergence curve, MRFO and GA trapped in local optima earlier than GMRFO. GMRFO produced more accurate solution which produce better step response for the system. The accuracy's mean for the three number of runs are 1.83, 1.99 and 1.77 for MRFO, GA and GMRFO respectively. Even thought these numbers have small different, however they produced a very significant different to their performance on the IPS.

Figure 10 shows graphical representation of the position and angle response. It is noticeable that, the rise time of step-response for all tuned controllers were almost the same. However, the position tuning from GMFRO is more accurate and stable while the pendulum angle settled at 0-radian. In contrast, MRFO and GA tuned controller did not settle at the correct position as well as the pendulum swung-up at 0-radian.

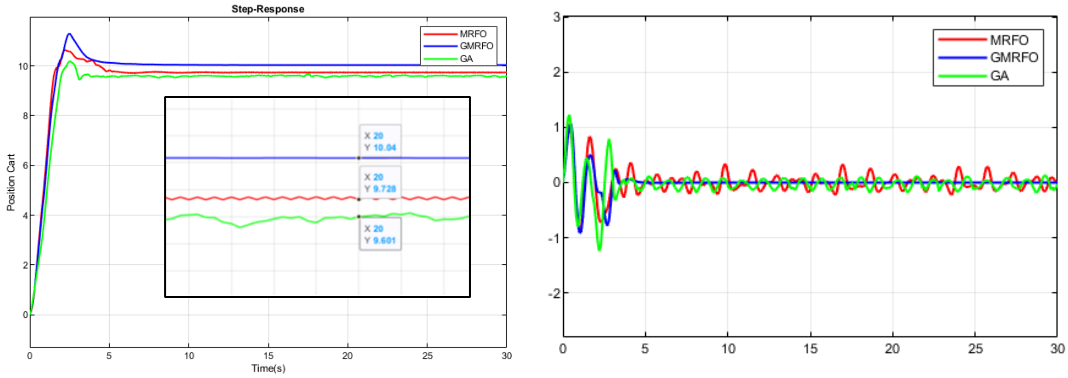

Figure 10. Step-response for cart position and pendulum angle for 30 seconds of simulation.

Table 4. Step response analysis.

Algorithm	MRFO	GA	GMRFO
Settling time, $\boldsymbol{t_s}$ (secs)	8.7	8.4	**8.1**
Rise time, $\boldsymbol{t_r}$ (secs)	**1.9**	2.4	2.0
Percent Overshoot, % OS	**17.3**	24.1	28.1
Steady state error, $\boldsymbol{e_{ss}}$	-0.272	-0.399	**0.04**

Table 4 shows the step response analysis of the tuned-controller. From the table, GMRFO can provide fastest settling time (8.1seconds) at the desired position. GA and MFRO came second and third respectively. At the meanwhile, MRFO still can provide fastest rise time (1.9seconds) and lowest percent overshoot (17.3%). However, MRFO (9.728cm) and GA (9.601cm) cannot be at the desired position as well as the pendulum stood-up at 0-radian. GMRFO, the most important outcome in this study, can provide the most accurate position (10.04cm) and stable for 30 seconds of simulation time. Therefore, it can be concluded that, GMRFO outperformed these MRFO and GA to provide desired cart position and 0-radian at the same time.

6. Conclusion

A new variant of MRFO called Hybrid-Genetic Manta Ray Foraging Optimization (GMRFO) algorithm has been presented in this paper. Components of Genetic Algorithm (GA) which are mutation and crossover operator were incorporated into MRFO to maximize the ability of searching strategy of the algorithm. Mutation and crossover were used to create more randomness in the algorithm by modifying location information of random agents. Mutation has improved the distribution of the searching agents of algorithm while crossover has improved the convergence rate of the algorithm. Thus, the chance to get a higher accuracy solution is increasing. To validate the performance, this proposed algorithm alongside with its parent MRFO and GA were used on 4 different benchmark functions. The averages were calculated alongside using Friedman and Wilcoxon sign rank test to determine the significant improvement of the algorithm. All algorithms also then furthered applied to optimize IT2FLC to control an Inverted Pendulum System (IPS). From the results, it can conclude that accuracy of solution produced by GMRFO outperformed MRFO and GA significantly. About the IT2FLC tuned by all algorithms, the result has shown that the GMRFO provided better tuned parameters. This can be observed by looking control response produced by the IPS. In the future, some variants of GMRFO can be developed. This proposed algorithm shown that it has potential to be improved further and can be used to solve a neural network model of robot system.

Acknowledgement
This work has been financially supported by the Fundamental Research Grant Scheme (FRGS/1/2019/ICT02/UMP/02/10) with a university reference number RDU1901196. It has been awarded by the Ministry of Higher Education Malaysia via Research and Innovation Department, Universiti Malaysia Pahang, Malaysia.

Reference

[1] W. Zhao, Z. Zhang and L. Wang, "Manta ray foraging optimization: An effective bio-inspired optimizer for engineering applications", Engineering Applications of Artificial Intelligence, vol. 87, p. 103300, 2020.

[2] Kitchen-Wheeler, Anne-Marie. “Visual identification of individual manta ray (Manta alfredi) in the Maldives Islands, Western Indian Ocean. Marine Biology Research”. 6. p. 351-363. 2010.

[3] L. Lie, "Heuristic Artificial Intelligent Algorithm for Genetic Algorithm", Key Engineering Materials, vol. 439-440, pp. 516-521, 2010.

[4] I. Rejer, "Classic genetic algorithm vs. genetic algorithm with aggressive mutation for feature selection for a brain-computer interface", Przegl.D Elektrotechniczny, vol. 1, no. 2, pp. 100-104, 2015.

[5] G. Geethamani and D. Mayilvaganan, "Analysis Of Genetic Crossover Techniques Based On Roulette Wheel Selection Algorithm And Steady State Selection Algorithm", International Journal Of Engineering And Computer Science, 2016.

[6] Ü. Önen, "Particle Swarm Optimization Based LQR Control of an Inverted Pendulum", Engineering and Technology Journal, vol. 2, no. 5, 2017.

[7] Liang, Jing & Qu, B. & Suganthan, Ponnuthurai. “Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization”. 2013.

[8] Chia-Feng Juang and Po-Han Chang, "Designing Fuzzy-Rule-Based Systems Using Continuous Ant-Colony Optimization", IEEE Transactions on Fuzzy Systems, vol. 18, no. 1, pp. 138-149, 2010.

[9] J. Mendel and D. Wu, "Critique of “A New Look at Type-2 Fuzzy Sets and Type-2 Fuzzy Logic Systems”", IEEE Transactions on Fuzzy Systems, vol. 25, no. 3, pp. 725-727, 2017.

[10] C. Ulu, M. Güzelkaya and I. Eksin, "Granular type-2 membership functions: A new approach to formation of footprint of uncertainty in type-2 fuzzy sets", Applied Soft Computing, vol. 13, no. 8, pp. 3713-3728, 2013.

[11] T. Harris and J. Hardin, "Exact Wilcoxon Signed-Rank and Wilcoxon Mann–Whitney Ranksum Tests", The Stata Journal: Promoting communications on statistics and Stata, vol. 13, no. 2, pp. 337-343, 2013.

[12] Abuelenin, Sherif & Abdel-kader, Rabab. (2017). Closed-Form Mathematical Representations of Interval Type-2 Fuzzy Logic Systems.

[13] Li, Zixiang & Kucukkoc, Ibrahim & Janardhanan, Mukund Nilakantan. (2017). ANOVA and Friedman Test Results.