IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 24 August 2022, accepted 12 September 2022, date of publication 20 September 2022, date of current version 20 October 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3208169

RESEARCH ARTICLE

A Hybrid Sperm Swarm Optimization and Genetic Algorithm for Unimodal and Multimodal Optimization Problems

BRYAN RAJ[1], ISMAIL AHMEDY[1,2], (Member, IEEE), MOHD YAMANI IDNA IDRIS[1,2], (Member, IEEE), AND RAFIDAH MD. NOOR[1,2]

[1]Department of Computer System and Technology, Faculty of Computer Science and Information Technology, Universiti Malaya, Kuala Lumpur 50603, Malaysia
[2]Centre for Mobile Cloud Computing, Faculty of Computer Science and Information Technology, Universiti Malaya, Kuala Lumpur 50603, Malaysia

Corresponding author: Ismail Ahmedy (ismailahmedy@um.edu.my)

This work was supported in part by the Ministry of Higher Education Malaysia Fundamental Research Grant Scheme (FRGS) under Grant FRGS/1/2019/ICT03/UM/02/2.

ABSTRACT A good exploration ability can ensure that the method jumps out of local optimum in multimodal problems and a good exploitation can ensure an algorithm converge faster to global optimum values. So, this paper proposes a new hybrid sperm swarm optimization and genetic algorithm to obtain global optimal solutions termed HSSOGA which is developed based on the concept of balancing the exploration and exploitation capability by merging Sperm Swarm Optimization (SSO), which has a fast convergence rate, and a Genetic Algorithm (GA) that can explore a search domain efficiently. To ensure that the proposed method delivers good performance, it is evaluated with 11 standard test function problems consisting of 5 unimodal and 6 multimodal functions. The proposed HSSOGA set is compared with conventional GA and SSO methods, as well as with several hybrid methods such as Hybrid Firefly and Particle Swarm Optimization (HFPSO), hybrid Simulated Annealing and Genetic Algorithm (SAGA), Hybrid Particle Swarm Optimization and Genetic Algorithm (HFPSO), hybrid Particle Swarm Optimization and Grey Wolf Optimization (PSOGWO), and closely related Hybrid Sperm Swarm Optimization and Gravitational Search Algorithm (HSSOGSA). The results are evaluated in terms of each method's best fitness, mean, standard deviation, and convergence rates. The numerical experiment results show that HSSOGA has better convergence towards the true global optimum values as compared to the conventional and existing hybrid methods in most unimodal and multimodal test function problems.

INDEX TERMS Algorithm, genetic algorithm, hybrid, metaheuristic, multimodal, optimization, sperm swarm optimization, unimodal.

I. INTRODUCTION

For the past decade, metaheuristic optimization approaches have been developed and used in engineering [1], [2], Wireless Sensor Networks (WSN) [3], and healthcare fields [4], [5]. Metaheuristics is widely used to enhance convergence rate and obtain global optimum solutions, as it promotes an exploration and exploitation capability. Exploration is the ability to explore different regions of the global search space that contain optimal solutions. Meanwhile, exploitation is the ability to focus on local regions that are identified by exploration, to obtain current optimal solutions [6], [7], [8]. Therefore, the main aim of metaheuristic approaches is to achieve a balance between the exploration and exploitation abilities to obtain global optimum solutions rather than a local optimum solution in a domain of search space.

However, the fundamentals of metaheuristic approaches can still be limited for a large computational problem with multiple local optima. Many hybrid metaheuristic approaches have been developed to find optimal solutions for large,

The associate editor coordinating the review of this manuscript and approving it for publication was Jamshid Aghaei.

multimodal real-life problems, to mitigate the issue. Metaheuristic algorithms are merged to add the method in finding a global optimal solution efficiently, as it balances out the exploration and exploitation capabilities. As such, several hybrid metaheuristic methods are explained and discussed below.

Selecting appropriate metaheuristics to be hybridized is a step that must be given close attention to, as it contributes to a good balance between exploration and exploitation, and also strengthens each other's weaknesses. In this study, the Sperm Swarm Optimization (SSO) algorithm and Genetic Algorithm (GA) have been selected to be hybridized to achieve a global optimal solution without drifting away towards local optima in both unimodal and multimodal test functions.

The main contributions of this paper are:

- Including crossover and mutation operators into a sperm swarm algorithm, thereby ensuring a balanced exploration and exploitation method to solve optimization problems.
- Applying velocity and position limits to the sperm's motility to avoid the method from drifting away from global optimal solutions.
- Evaluating the proposed method with existing hybrid methods that are widely used in solving real life problems to ensure that the proposed method can solve unimodal and multimodal problems efficiently by obtaining best fitness.
- Discussing the advantages and disadvantages of the proposed method and its future direction in real life optimization problems.

The remainder of the paper is arranged as follows. Section 2 presents the literature review. Section 3 presents the methodology of the method. Section 4 presents details on the proposed hybrid sperm swarm optimization and genetic algorithm (HSSOGA). Section 5 presents the experimental information. Section 6 presents results obtained from the evaluation. Section 7 presents a discussion of the obtained results, and Section 8 presents a conclusion to the article with future suggestions.

II. LITERATURE REVIEW

Metaheuristics is a high-level algorithmic framework that was designed to tackle complex optimization problems which are continuous, discrete, unconstrained, and multi-objective in nature [9]. They are extended heuristics that improve the exploration and exploitation capabilities [8]. Metaheuristic approaches can be classified into several groups, which are, swarm-based metaheuristics, evolutionary-based metaheuristics, physics-based metaheuristics, and human-based metaheuristics [10], [11]. Swarm-based metaheuristics are mostly inspired by behaviours of animals and plants which are capable of interacting with each other and with the environment. Examples of swarm-based metaheuristics are "Particle Swarm Optimization (PSO)" [12], "Sperm Swarm Optimization (SSO)" [13], and "Grey Wolf Optimizer (GWO)" [14]. Evolutionary-based metaheuristics are the techniques developed based on the laws of natural evolution. The most common evolutionary-based metaheuristics are "Differential Evolution (DE)" [15] and "Genetic Algorithms (GA)" [16]. Physics-based metaheuristics are inspired by the laws of physics in the universe. "Gravitational Search Algorithm (GSA)" [17] and "Simulated Annealing (SA)" [18] are examples of metaheuristic methods that are created based on the principles of physics. Human-based approaches on the other hand, are developed based on the behaviours, characteristics, and lifestyles of human beings. Examples of human-based approaches are "Harmony Search Algorithm (HS)" [19] and "Human Group Formation (HGF)" [20].

Furthermore, metaheuristic methods can also be classified into competitive and cooperative metaheuristic optimizers. In the year 2016, a competitive optimization algorithm based on the competitive behaviour of various creatures in nature was proposed [44]. The work tends to make few algorithms compete among each other to solve an optimization problem which concludes that different optimization problem needs different optimization algorithm. So, it can be said that competitive methods are methods that compete within the metaheuristic methods to achieve its goal which are mostly non-hybrid algorithms. For example, SSO is a competitive method where in SSO the sperms compete to reach the ovum in an optimal way [13]. In a recent work, a cooperative metaheuristic algorithm for global optimization problems was proposed [45]. In that work, operators of six metaheuristic methods are combined to solve global optimization problems termed SSIO. The proposed method seems to perform better than other state-of-art that it was compared to. This shows that the cooperation of few operators from different algorithm can ensure the increase in diversity and better convergence rate. So, cooperative metaheuristics are mostly comprised of hybrid algorithms that uses various metaheuristic functions in one algorithm such as HFPSO, SAGA, HPSOGA, HGASSO, PSOGWO and HSSOGSA which is discussed below.

A very good example of the cooperative hybrid method is the hybrid firefly and particle swarm optimization (HFPSO) algorithm which was introduced by [21] and is used to obtain global solutions for computationally expensive numerical problems. The author's motive was to combine the strengths of both algorithms in obtaining a method with the balance of exploration and exploitation capabilities. In this paper, the author uses PSO for global searching as it is deemed to have fast convergence, and FA for local searching as it fine-tunes the exploitation. The developed hybrid method was then evaluated by using "congress on evolutionary computation (CEC)" CEC 2015 and CEC 2017 benchmark functions consisting of unimodal, simple multimodal, hybrid, and composition functions. The results showed that the proposed HFPSO method performs much better than standard PSO, FA and other hybrid methods. The disadvantage of this hybrid method is that it might not have the best balance between exploration and exploitation, as swarm-based algorithms are known for better exploitation as opposed to exploration. In addition, [46] suggests that hybrid of swarm optimization (HSO) has to be combined with different algorithm because HSO has the ability to reach suboptimal solutions frequently. So, selecting two appropriate method for hybridizing is vital for performance efficiency.

In an effort to obtain a balance between exploration and exploitation, [22] proposed a combined simulated annealing and genetic algorithm (SAGA) to improve network lifetime and energy efficiency in mobile ad-hoc networks (MANETs). The merging of SA and GA focuses on overcoming the large combinational optimization problems so that a greater performance of MANET can be achieved. SAGA was compared with memory enhanced genetic algorithm (MEGA) and the results showed that SAGA outperforms MEGA in terms of lesser energy consumption and increased network lifetime of MANET. However, both GA and SA are well known for their exploration capabilities, limiting global and local search ability [7].

The overwhelming use of hybrid algorithms inspired [23] to propose a new hybrid particle swarm optimization and grey wolf optimization (HPSOGWO) algorithm to obtain global solutions by having balanced exploration and exploitation capabilities. The authors stated that by using the GWO algorithm, the possibility of PSO falling into the local minimum is reduced. The developed hybrid method was evaluated on 5 benchmark functions and 3 real-world problems that consisted of parameter estimation for frequency-modulated sound waves, process flow sheeting problem, and leather nesting problem (LNP). The results showed that HPSOGWO performs better than the standard PSO, GWO, ABC, SSA, and three hybrid PSO–GWO approaches in terms of converging to lower-cost values with fewer iterations. However, the time complexity of HPSOGWO is higher as compared to standard PSO and GWO, but the authors' concern was towards getting higher performance.

The fast convergence nature of PSO and the good global searching ability of GA inspired [24] to propose a hybrid particle swarm optimization and genetic algorithm (HPSOGA). Since PSO is usually easy to be trapped in local optimum, the author integrated the genetic operator of GA, which consists of crossovers and mutations, into PSO, which promotes a good balance between exploration and exploitation capabilities. The proposed algorithm was evaluated and tested with engineering constrained optimization problems such as pressure vessel design and welded beam design. The results showed that the proposed method is more effective and robust compared to the conventional GA and PSO algorithms.

In very recent research, [25] proposed a hybrid sperm swarm optimization and gravitational search algorithm (HSSOGSA) to ensure a good balance between exploration and exploitation capabilities for global optimization. SSO seemed to outperform the well-known PSO in obtaining the global solutions, which motivated the author to combine the capability of exploitation in SSO with the capability of exploration in GSA. This combination of SSO and GSA was done using a co-evolutionary heterogeneous low-level hybrid technique, as both approaches run simultaneously, which reduces the time complexity of the method. To evaluate the efficiency and performance of the proposed method, the author tested HSSOGSA under different testbed problems of optimization called the CEC 2017 suite. The results described that the proposed method has greater performance to jump out of local extremes with a faster rate of convergence compared to the standard SSO and GSA methods in most of the CEC 2017 suite benchmark functions.

To promote better balance between exploration and exploitation, [26] proposed an improved memetic method for clustering to balance the node's load in WSNs. Memetic algorithms can be said as improved GA where it is hybridized with local search ability. Since GA has a limitation of slow convergence rate, researchers use local search such as hill climbing, simulated annealing or tabu search methods to enhance the overall algorithm to reach optimized solution faster and efficiently [27], [28]. So, to ensure that the method first explore the search space, population generation, selection, crossover, and mutation phases are carried out followed by a local search to ensure optimized solution from exploration. The authors evaluated the proposed method with two popular predecessor method which are GA and Low-energy adaptive clustering hierarchy (LEACH). The results shows that the proposed method outperformed LEACH by great margin and performed slightly better than GA. This is because the proposed method has similar phases to GA and the local search method does not promotes significant exploitation for better results.

Since certain algorithms contributes hugely to a good balance between exploration and exploitation, [29] recently proposed a Hybrid Genetic Algorithm and Sperm Swarm Optimization (HGASSO) to optimize multimodal functions. The authors applied local search which is SSO first to select the global best solution and personal best solution before the selection, crossover and mutation is applied to jump out of the local minima easily. The method was tested with 11 multimodal minimization test functions, and it is compared with the conventional SSO and GA. From the results, the proposed hybrid method outperformed the conventional methods in 6 out of 11 test function, performed the same with SSO in 2 out of 11 test functions and performed equally to the conventional methods in 3 out of 11 test functions. The authors also used One-way ANOVA (Tukey's test) to determine the significance of the results to justify the performance of proposed method. Even though, the results seem convincing, the authors did not ensure the performance of the method in unimodal optimizations as some hybrid algorithms might not perform well in unimodal functions. Moreover, the comparison between existing hybrid algorithms that limits the justification upon the performance of the algorithm. The strengths and weaknesses of HGASSO are tabulates as below.

Being motivated upon the strengths HGASSO and the hybridization potential of other existing hybrid methods, we propose a Hybrid Sperm Swarm Optimization and Genetic Algorithm (HSSOGA). To ensure that the weaknesses of HGASSO is mitigated, we adopt the memetic algorithm

TABLE 1. **Strengths and weaknesses of HGASSO.**

Strengths	Weaknesses
Hybridizes GA which is a good exploration-based algorithm with SSO which is a good exploitation-based algorithm [31].	Appling SSO first will initialize the sperms to exploit a certain region before exploring to other region which may cause the algorithm to be stuck in local optimum is high multimodal problems.
Uses a good set of multimodal test function to test its performance compared to its conventional method.	Does not evaluate the method with unimodal test functions and does not compare the results with existing hybrid algorithms to ensure the full potential of the HGASSO.

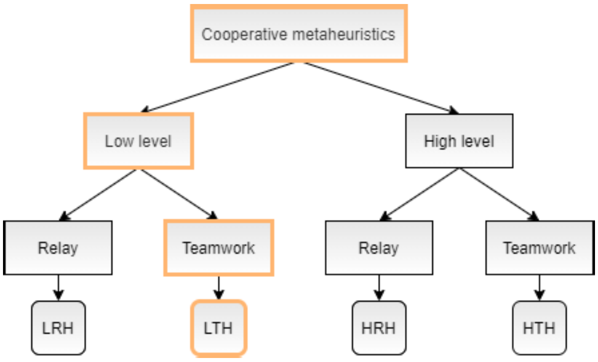


FIGURE 1. **The four classes of cooperative metaheuristics: LRH, LTH, HRH, HTH.**

technique by ensuring the individuals to explore the search space first broadly to skip the local optima using GA first and then exploit the searched space for optimized solutions using SSO (local search). This is because, the exploitation is said to be optimized from the outcome of exploration [30]. The Sperm swarm optimization (SSO) algorithm is one of the swarm-based methods that was recently developed; it focuses on the behaviour of sperm cells on natural fertilization [31]. SSO is deemed to produce better solutions and outperform PSO, which is widely used in literature for its local search capability (exploitation) in several benchmark tests [31]. This shows that SSO is an important candidate for hybridization. On the other hand, GA is an algorithm that is developed based on Charles Darwin's theory of natural evolution, where it uses selection, crossover, and mutation operators to ensure diversity in the search space [16]. GA, which has been remarkably used for the past decade, shows that it has a high-quality global searching ability (exploration) to obtain global solutions [32], [33].

Some advantages of SSO and GA are stated below:

- SSO has good exploitation capability.
- SSO is easy to implement and understand.
- SSO does not have too many calculations.
- SSO has a greater ability to find the optimal results in a unimodal function.
- GA has good exploration capability.
- GA has been widely used in many applications successfully.
- GA is able to find a better global optimum in multimodal functions.

Some limitations of SSO and GA are stated below:

- SSO might fall into local optima (limitation on exploration)
- GA can be slow in convergence (limitation on exploitation)

To ensure that the limitations are mitigated, we are developing a hybrid approach by merging SSO and GA methods, so that a balance between exploration of GA and exploitation of SSO is achieved, to obtain a global optimum solution. SSO, which has poor global search capability, which makes it fall into local optimum, will capitalize on GA's diverse global search capability to reach global regions, while GA, which is slow in convergence, will capitalize SSOs fast convergence nature to search locally for optimal solutions. To ensure that our method performs well, several unimodal and multimodal test functions from the "Virtual Library of Simulation Experiments" [34] are used to compare the performance of the proposed method with conventional SSO and GA, as well as several hybrid methods from the literature. The motivations of this research are simplified as below:

- To develop a method with balanced exploration and exploitations capability to solve both unimodal and multimodal problem efficiently.
- To mitigate the conventional algorithm's drawback successfully by hybridization.
- To ensure faster convergence rate by adopting the memetic algorithm technique in hybridization.
- To ensure the developed modelled are validated and evaluated with the comparisons of existing hybrid algorithms to have a solid justification towards the performance of the proposed method.

III. METHODOLOGY

Since hybridization consists of two conventional algorithms merging to form a method, this section will briefly describe the inspiration, flow, structure, characteristics, and mathematical modelling of both, the conventional SSO and GA algorithms.

Hybrid metaheuristics has become a pivotal approach in solving optimization problems as it promotes a balance between exploration and exploitation. Moreover, a hybrid metaheuristic also reduces the limitation of conventional algorithms, where both the algorithms try to cancel out each other's limitations. Hybridization can be done through several methods as depicted in Figure 1 [35].

Cooperative metaheuristic is designed based on 2 phases which are low-level/high-level and relay/teamwork. The usage of low-level optimization in the first phase is a composite of single optimization method where a metaheuristic is changed by another metaheuristic. On the other hand, then high-level optimization is a metaheuristic that has no direct relationship to its internal processes. In the second phase, using relay hybridization means the metaheuristics are applied one after another creating a sequential process where each uses the previous output as its input. The teamwork hybridization on the other hand has many parallel cooperating agents where each agents makes its search in the search domain [35], [43]. So, using low-level optimization in phase 1 and relay hybridization in phase 2 creates Low-level Relay Hybrid (LRH) which is a method is embedded into another method and executed sequentially. Utilizing low-level optimization with teamwork hybridization creates Low-level Teamwork Hybrid (LTH) where the element of a method is replaced in another method, and it performs parallelly. Moreover, usage of high-level optimization with relay hybridization yields High-level Relay Hybrid (HRH) where two methods are self-contained and executed sequentially. The last class is called High-level Teamwork Hybrid (HTH) which utilizes the high-level optimization of phase 1 with teamwork hybridization of phase 2. HTH contained methods that are self-contained and works parallelly [35]. So, in this paper, we hybridize SSO and GA using the LTH method where it allows a method to be embedded into a global method and executed in parallel. Based on [35], this hybrid method can also be called ‘‘Parallel Collaborative Hybrids’’, where two algorithms are run simultaneously by changing the same population.

A. CONVENTIONAL GENETIC ALGORITHM (GA)

For the past decade, a plethora of research has used GA as an optimization method for multiple applications such as power electronics, wireless sensor networks, and airline bookings [36], [37], [38]. GA is an algorithm developed by [16] based on Charles Darwin’s theory of survival of the fittest, where it is a biological evolution process. GA initially starts with a population consisting of random chromosomes that are later selected to apply crossover and mutation.

Crossover operators exchange some genes in a specific way from the selected chromosomes that act as the parents to generate new offspring (new solutions) [39]. A Uniform crossover operator is adopted in this paper. Uniform crossovers have the advantage of unbiased exploration, and they are applicable to be used on large subsets. However, they produce less diverse solutions.

Mutation operators are used to maintain the diversity of individuals (solutions) from one population to the next population, so that the solutions don’t get trapped into local optima. The mutation operator works by changing some genes from an individual chromosome, which then results in it carrying different characteristics from their parents (diverse solution) [39].

B. CONVENTIONAL SPERM SWARM OPTIMIZATION (SSO)

Sperm swarm optimization was proposed recently by [13] for wireless sensor network (WSN) challenges. The algorithm was inspired from the natural fertilization process, where a swarm of sperm cells swim towards the ovum to merge with it. During this process, only one out of millions of sperm cells is the winner. In the beginning, the swarm of sperm cells reside in the cervix randomly with two velocities on X-axis and Y-axis.

The behaviour of a swarm of sperm cells swimming towards the ovum exhibits a behaviour similar to ‘‘flocking’’. The movement of sperm is affected by two important parameters which are pH value and temperature inside the female reproductive system. These two parameters define the sperm’s motility and movement direction. According to the findings in [40], the pH value in a female reproductive system is around 4.5 to 5.5, while the temperature inside a female reproductive system can vary between 35.1 °C to 37.4 °C. However, [13] states that the alkaline pH value, which is around 7 to 14, is the most suitable for the sperm’s movement, and the temperature in a female reproductive system may go up to 38.5 °C because of the vaginal blood pressure.

To translate this phenomenon in an optimization environment, the sperm cells act as a candidate solution that moves in a multidimensional search space domain to obtain a global optimal solution. The swarms also record the best solution in their tracks, which means that the optimal sperm (from the global optimal sole parent sperm cells that were successful in fertilizing the ovum) and the local optimal solution (sperm optimal solution) are considered.

IV. PROPOSED HYBRID SPERM SWARM OPTIMIZATION AND GENETIC ALGORITHM (HSSOGA)

A. INITIALIZATION

Initially, all the sperm cells are randomly positioned using continuous uniform distribution, where each sperm represents a candidate solution. The initial fitness of the population is evaluated and sorted. In this process the global best, x_{sgBest} is also updated after the initial evaluation, to set as a benchmark for the iterations that follow.

B. SELECTION

In this process, two sperms are selected from the initial population using the ‘‘Roulette Wheel’’ technique, where all the possible chromosomes are attached to the wheel and the wheel is rotated randomly to select specific chromosomes for the crossover and mutation process [40]. The probability, $Prob_i$ of selecting specific individuals using roulette wheel selection is expressed in Eq. (1) and Eq. (2).

$$Prob_i = \exp\left(\frac{-beta\ .\ Fit_i}{WorstFit_i}\right) \tag{1}$$

$$Prob_i = \frac{Prob_i}{\sum_{i=1}^{nPop} Fit_i} \tag{2}$$

where selection pressure, $beta = 8$, Fit_i is the fitness of the chromosome, $WorstFit_i$ is the worst fitness obtained, and $nPop$ is the size of the population.

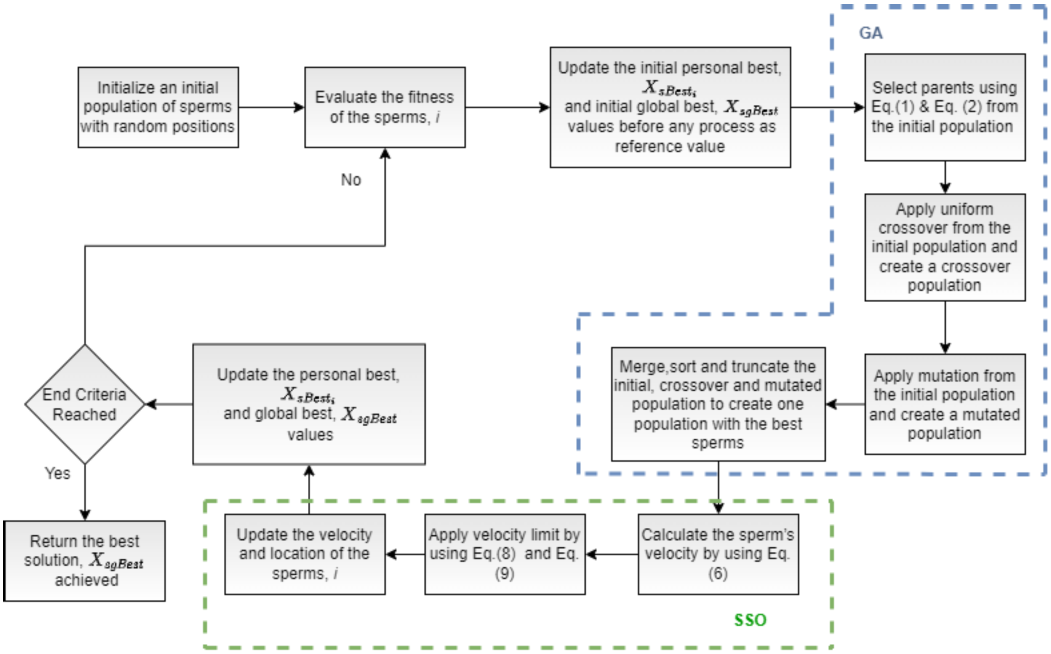

Initialize an initial population of sperms with random positions
Evaluate the fitness of the sperms, i
Update the initial personal best, X_{sBest_i} and initial global best, X_{sgBest} values before any process as reference value
GA
Select parents using Eq.(1) & Eq. (2) from the initial population
Apply uniform crossover from the initial population and create a crossover population
Apply mutation from the initial population and create a mutated population
Merge,sort and truncate the initial, crossover and mutated population to create one population with the best sperms
No
End Criteria Reached
Yes
Update the personal best, X_{sBest_i} and global best, X_{sgBest} values
Return the best solution, X_{sgBest} achieved
Update the velocity and location of the sperms, i
Apply velocity limit by using Eq.(8) and Eq. (9)
Calculate the sperm's velocity by using Eq. (6)
SSO

FIGURE 2. The process flow of the proposed HSSOGA.

C. CROSSOVER AND MUTATION

Upon selecting the sperm, the crossover process begins, which ends up producing a new population called crossover population. Following this, the mutation process begins a mutation of the sperm cells from the initial population, producing another new mutated population.

D. MERGE, SORT AND TRUNCATE

The populations from crossover and mutation processes are merged and sorted in ascending order of the values. It is then truncated to the number of populations, *nPop*, set at the beginning of the method, to ensure that the best population is obtained.

E. VELOCITY AND POSITION UPDATE

The initial sperm velocity, V_0 is calculated using Eq. (3).

$$V_0 = Damp \ . \ V_i \ . \ \log_{10}(pH_1) \tag{3}$$

where $Damp$ is the damping factor (0 to 1), V_i is current sperm velocity, and pH_1 is a random pH value between 7 and 14.

The personal best solution is expressed by Eq. (4), and the global best solution is expressed by Eq. (5).

$$CurrentBestSol(t) = \log_{10}(pH_2) \ . \ \log_{10}(Temp_1) \ . \ (x_{sBest_i} - x_i(t)) \tag{4}$$

$$GlobalBestSol(t) = \log_{10}(pH_3) \ . \ \log_{10}(Temp_2) \ . \ (x_{sgBest} - x_i(t)) \tag{5}$$

where pH_2 and pH_3 are random pH values between 7 and 14, $Temp_1$ and $Temp_2$ are random temperature values between 35.1 °C and 38.5 °C, x_{sBest_i} is the personal best location of sperm i at iteration t, x_{sgBest} is the global best location of the sperm (global optimal solution), and x_i is the current location of the sperm at iteration t.

The velocity of the sperm, V_i is evaluated as per Eq. (6).

$$\begin{aligned} V_i = {} & Damp \ . \ V_i \ . \ \log_{10}(pH_1) \\ & + Log_1 0(pH_2) \ . \ \log_{10}(Temp_1) \ . \ (x_{sBest_i} - x_i(t)) \\ & + \log_{10}(pH_3) \ . \ \log_{10}(Temp_2) \ . \ (x_{sgBest} - x_i(t)) \end{aligned} \tag{6}$$

The current position of the sperm (current solution) is calculated as depicted in Eq. (7), to ensure that the position updates on each iteration towards achieving the global optimal solution.

$$x_i(t) = x_i(t) + v_i(t) \tag{7}$$

To avoid the method from drifting away from the global optima solution, velocity limits and position limits are applied before evaluating the fitness. The maximum and minimum velocities are calculated in Eq. (8) and Eq. (9).

$$V_{max} = 0.1 * (Var_{max} - Var_{min}) \tag{8}$$

$$V_{min} = -V_{max} \tag{9}$$

where V_{max} is the maximum velocity limit and V_{min} is the minimum velocity limit, and Var_{max} and Var_{min} are the maximum and minimum position limits, respectively. In other words, Var_{max} and Var_{min} are the maximum and minimum values of search domains.

Upon completing the velocity and position update process, the population is then merged, sorted, and truncated again for the next iteration. The fitness of the population is then evaluated and updated to see if the values achieved are better than the previous global best solution.

The flow of the overall process of HSSOGA is described in Figure 2.

The overall algorithm of Hybrid Sperm Swarm Optimization and Genetic Algorithm (HSSOGA) is depicted by the pseudocode below:

Algorithm 1 Hybrid Sperm Swarm Optimization and Genetic Algorithm (HSSOGA)

```
Begin
Step 1: Set the number of population (nPop),maximum iteration (MaxIter) and iter=0.
Step 2: Initialize the population(sperm,i)and calculate the fitness.
Step 3: while (iter < MaxIter)
Step 4: Calculate selection probabilities by using Eq.(1) and Eq.(2).
Step 5: Use Roulette Wheel to SELECT parents.
Step 6: Use Uniform Crossover on the selected parents.
Step 7: Use Mutation.
Step 8: Merge,sort and truncate the population
Step 9: Apply SSO
    for i=1 : population size do
        Calculate the sperm's velocity by using Eq.(6)
        Apply velocity limit by using Eq.(8) and Eq.(9)
        Update the position of the sperm by using Eq.(7)
        Apply position limit
    end for
Step 10: Obtain the $x_{sgBest}$ value.
End
```

V. EXPERIMENTAL SETTINGS

The main purpose of this work is to ensure a good balance between the exploration and exploitation capabilities, and the good performance of the proposed method. The proposed HSSOGA is evaluated with 11 test problems that further consist of 5 unimodal and 6 multimodal test problems. The mathematical notion, their range of search space domain and their dimensions are described in Appendix A. The results are compared and tabulated into 2 categories. Firstly, we evaluate and compare the proposed HSSOGA with the conventional SSO and GA. Secondly, we evaluate and compare the proposed HSSOGA with existing hybrid methods such as HFPSO, SAGA, HPSOGA, PSOGWO, and HSSOGSA. All the methods are programmed in MATLAB R2021a on a computer running Windows 10 Pro with 16GB DDR4 RAM and an AMD Ryzen 5 5600X 6-Core 3.7 GHz processor.

The methods are run through the benchmark functions a total of 30 times to ensure the accuracy of the obtained results. All the functions that are described in Appendix A are minimization functions, where functions 1, 2, 3, 4, 5, 6, 7, 8, 9, and 11 have a minimum optimal value of 0, and function 10 has a minimum optimal value <-9.66015 [34].

To evaluate the performance and efficiency of the proposed HSSOGA with accuracy, mean, standard deviation, and best fitness criteria are used. These criteria are described below:

TABLE 2. List of parameters of SSO, GA, HSSOGA, HSSOGSA, HPSOGA, SAGA, HFPSO and PSOGWO.

Parameters	Values
HSSOGA	
Velocity damping factor (D)	Rand (0, 1)
Temperature	Rand (35.5, 38.5)
pH	Rand (7, 14)
Crossover Percentage (pc)	0.7
Mutation Percentage (pm)	0.3
Mutation Rate (mu)	0.1
SSO	
Velocity damping factor (D)	Rand (0, 1)
Temperature	Rand (35.5, 38.5)
pH	Rand (7, 14)
GA	
Crossover Percentage (pc)	0.7
Mutation Percentage (pm)	0.3
Mutation Rate (mu)	0.1
HSSOGSA	
Velocity damping factor (D)	Rand (0, 1)
Temperature	Rand (35.5, 38.5)
pH	Rand (7, 14)
α	20
G_0	1
HPSOGA	
Inertia Weight Damping $(wdamp)$	0.99
$c1$	1.5
$c2$	1.5
Crossover Percentage (pc)	0.7
Mutation Percentage (pm)	0.3
Mutation Rate (mu)	0.1
SAGA	
Initial temperature $(initTemp)$	1000
Final temperature $(finTemp)$	1
Crossover Percentage (pc)	0.7
Mutation Percentage (pm)	0.3
Mutation Rate (mu)	0.1
HFPSO	
$c1$	1.5
$c2$	1.5
Inertia weight damping factor (w)	0.9
α	0.2
B_0	2
γ	1
PSOGWO	
$c1, c2, c3$	0.5
$r1, r2, r3$	Rand (0,1)
Inertia weight (w)	0.7298

Mean (μ): Mean is used to find the average fitness values after running the method N times to ensure the accuracy of the fitness values obtained, as depicted in Eq. (10).

$$\mu = \frac{\sum_{i=1}^{n}(f_i)}{N} \tag{10}$$

Standard deviation (σ): Standard deviation is used to find the dispersion between the values of the fitness function after running the method for N times, as depicted in Eq. (11).

$$\mu = \sqrt{\frac{\sum_{i=1}^{n}(f_i - \mu)^2}{N-1}} \tag{11}$$

Best fitness (optimal value): Best fitness is obtained by finding the minimum fitness value achieved from running the

TABLE 3. The numerical comparison results with conventional methods.

Test Function		SSO	GA	HSSOGA
F1	**Best**	**1.74E-251**	1.91E-32	1.71E-160
	Average Best	**2.49E-204**	5.40E-06	1.08E-150
F2	**Best**	**3.10E-249**	7.90E-32	1.02E-161
	Average Best	**6.21E-193**	7.03E-05	2.41E-146
F3	**Best**	7.25E-80	2.297122519	**4.51E-157**
	Average Best	2.50E-72	25.4494108	**3.34E-147**
F4	**Best**	26.5203676	24.75827686	**22.84251707**
	Average Best	27.46019878	62.23417757	**23.46738764**
F5	**Best**	4.265736337	**0.0000**	**0.0000**
	Average Best	4.706287994	**8.57E-10**	8.28E-09
F6	**Best**	**0.0000**	**0.0000**	**0.0000**
	Average Best	**0.0000**	0.004446813	**0.0000**
F7	**Best**	**8.88E-16**	7.99E-15	**8.88E-16**
	Average Best	1.01E-15	4.40E-05	**8.88E-16**
F8	**Best**	**0.0000**	9.07E-06	**0.0000**
	Average Best	**0.0000**	0.003673853	**0.0000**
F9	**Best**	8642.733457	947.5072532	**947.5070587**
	Average Best	9569.462714	1654.189346	**1575.230715**
F10	**Best**	-6.297533967	**-29.3050437**	-28.27425892
	Average Best	-4.197229975	**-26.00042022**	-25.7070616
F11	**Best**	-45.29747145	**-45.29761135**	**-45.29761135**
	Average Best	-45.13970782	**-45.29761135**	**-45.29761135**

TABLE 4. The statistical comparison results with conventional methods.

Test Function	SSO		GA		HSSOGA	
	μ	σ	μ	σ	μ	σ
F1	135.429	2319.162	330.3047	2829.527	**72.48017**	**1195.676**
F2	2141.296	34324.59	4460.717	38324.1	**1197.833**	**20002.23**
F3	3073.699	97098.5	10134.05	315380.8	**4.749502**	**34.06526**
F4	290755	6615326	618321.2	8309928	**107015.4**	**2621449**
F5	114.5638	2011.76	330.0602	2859.767	**70.72267**	**1171.976**
F6	1.245399	21.82383	2.910744	23.14966	**0.733629**	**11.44752**
F7	0.245736	1.707526	0.609113	2.48301	**0.167571**	**1.341744**
F8	**2.867999**	**23.85509**	18.01666	45.59172	5.663088	29.45629
F9	8642.733	**1.46E-10**	1524.119	1552.319	**1474.353**	1439.745
F10	-6.29753	**4.53E-14**	**-26.353**	4.222277	-25.2374	4.416332
F11	-45.2934	0.057219	-45.2964	0.033638	**-45.2967**	**0.017091**

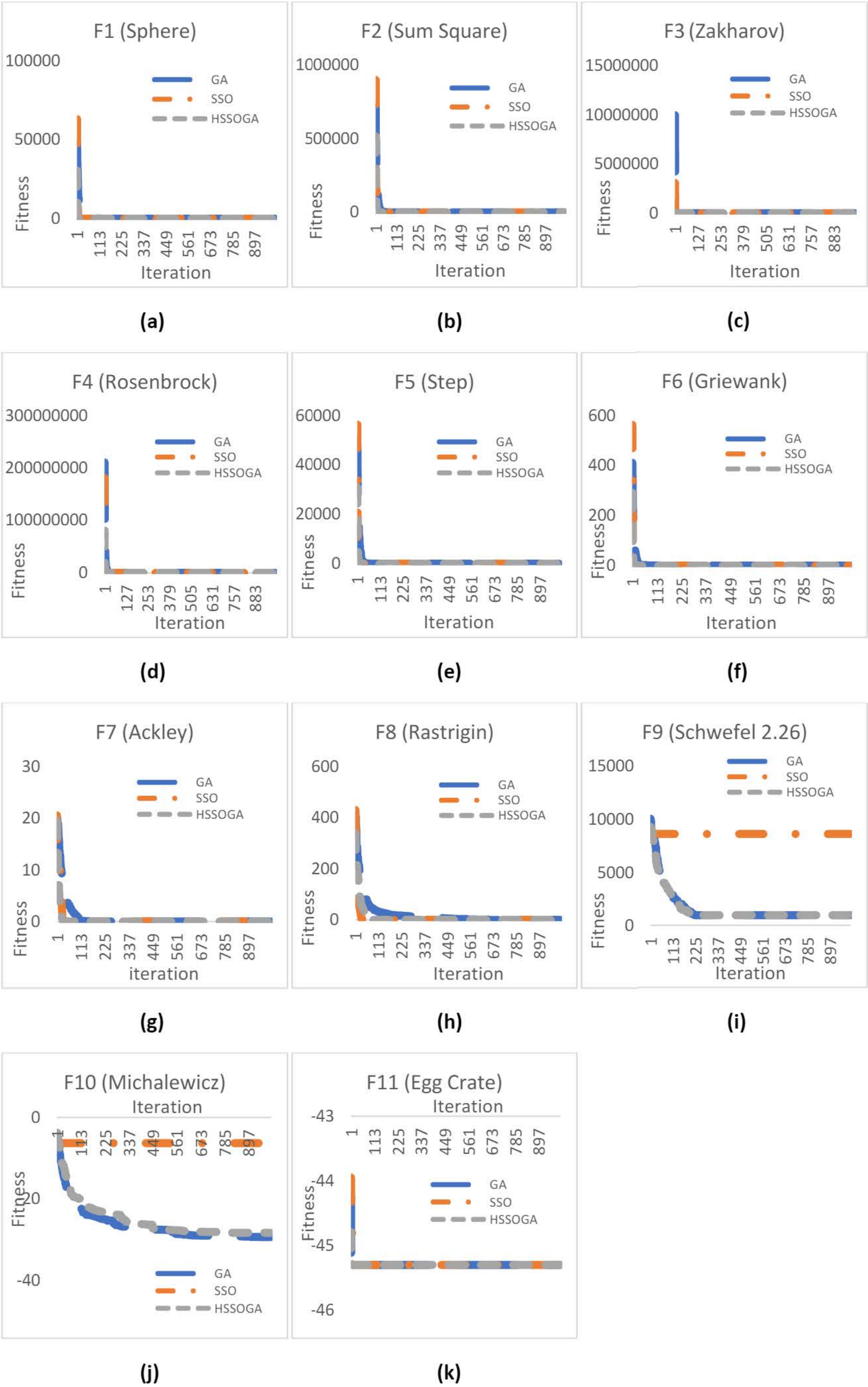
TABLE 5. Statistical analysis of results using "One-way ANOVA (Tukey's test)" between HSSOGA and the conventional methods.

Test Function	Algorithm (I)	Algorithm (J)	Mean Difference (I-J)	p-Value (Sig.)
F1	HSSOGA	GA	**-257.82457***	**0.026***
		SSO	-62.94885	0.802
F2	HSSOGA	GA	-3262.88436	0.057
		SSO	-943.46290	0.786
F3	HSSOGA	GA	-10129.2968	0.460
		SSO	-3068.9494	0.931
F4	HSSOGA	GA	-511305.798	0.166
		SSO	-183739.662	0.792
F5	HSSOGA	GA	**-259.33753***	**0.018***
		SSO	-43.84110	0.890
F6	HSSOGA	GA	**-2.17711***	**0.034***
		SSO	-0.51177	0.828
F7	HSSOGA	GA	**-0.44154***	**<0.001***
		SSO	-0.07816	0.629
F8	HSSOGA	GA	**-12.35357***	**<0.001***
		SSO	2.79509	0.161
F9	HSSOGA	GA	-49.76584	0.634
		SSO	**-7168.3801***	**<0.001***
F10	HSSOGA	GA	**1.11559***	**<0.001***
		SSO	**-18.93990***	**<0.001***
F11	HSSOGA	GA	-0.00038	0.975
		SSO	-0.00333	0.145
***. The mean difference is significant at the 0.05 level.**				

method N times, as depicted in Eq. (12).

$$Best_{fit} = \underset{1\leqq i\leqq N}{min(f_i)} \tag{12}$$

Average best fitness: Average best fitness is calculated by averaging the best fitness values over 30 independent runs, as depicted in Eq. (13). This metric is used to determine



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j) (k)

FIGURE 3. (a-k) Comparison of the convergence rate with conventional methods.

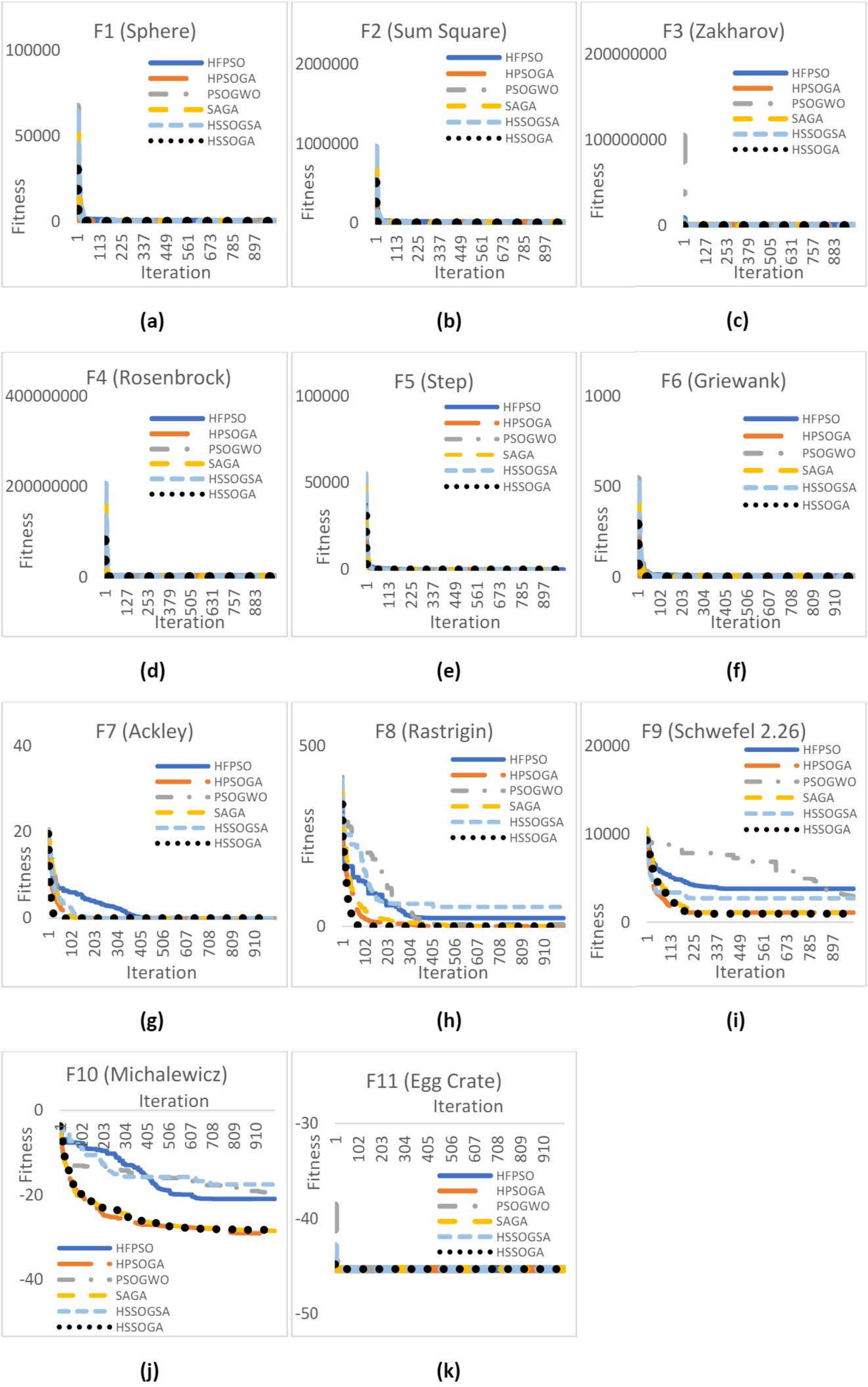


FIGURE 4. (a-k) Comparison of the convergence rate with existing hybrid methods.

TABLE 6. The numerical comparison results with existing hybrid methods.

Test Function		HFPSO	HPSOGA	SAGA	PSOGWO	HSSOGSA	HSSOGA
F1	**Best**	2.56E-16	7.93E-61	9.88E-34	1.80E-56	7.83E-19	**1.71E-160**
	Average Best	2.57E-15	2.53E-55	1.55E-06	467.7724477	1.26E-18	**1.08E-150**
F2	**Best**	3.80E-15	8.91E-60	8.06E-31	1.67E-50	1.35E-17	**1.02E-161**
	Average Best	5.03E-14	3.70E-54	2.61E-06	13430.95666	2.33E-17	**2.41E-146**
F3	**Best**	5.02E-06	1.25E-08	10.47547543	1.40E-33	4.33E-18	**4.51E-157**
	Average Best	0.000118845	1.01E-06	40.54598855	31.44696487	7.22E-18	**3.34E-147**
F4	**Best**	**9.748974018**	13.27780311	24.6857928	24.74482611	19.89568571	22.84251707
	Average Best	34.22436546	31.24158055	102.7739149	3978382.442	53.25170091	**23.46738764**
F5	**Best**	6.48E-17	**0.0000**	1.45E-31	1.67E-06	7.15E-19	**0.0000**
	Average Best	2.07E-15	**2.05E-33**	1.22E-05	1658.638777	1.27E-18	8.28E-09
F6	**Best**	5.55E-16	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**
	Average Best	0.008611388	0.001395855	0.002547124	22.63170768	0.018247602	**0.0000**
F7	**Best**	7.87E-09	7.99E-15	7.99E-15	2.58E-14	5.87E-10	**8.88E-16**
	Average Best	2.79E-08	1.90E-14	9.19E-06	3.139933571	0.075724197	**8.88E-16**
F8	**Best**	22.88401797	**0.0000**	5.85E-06	5.304967577	53.7276325	**0.0000**
	Average Best	44.40827769	9.56E-06	0.00232185	106.0308325	94.88555354	**0.0000**
F9	**Best**	3770.361081	1065.945393	1065.945393	2920.004998	2689.178397	**947.5070587**
	Average Best	5000.556758	**1547.594621**	1575.230519	5076.574427	3913.916748	1575.230715
F10	**Best**	-20.9167687	**-29.11364757**	-28.47474317	-19.50307577	-17.48954989	-28.27425892
	Average Best	-17.09591952	**-27.10084291**	-26.46601774	-13.25439421	-13.96881241	-25.7070616
F11	**Best**	**-45.29761135**	**-45.29761135**	**-45.29761135**	**-45.29761135**	**-45.29761135**	**-45.29761135**
	Average Best	**-45.29761135**	**-45.29761135**	-45.29760766	-45.29714592	**-45.29761135**	**-45.29761135**

the resistance of the method from being trapped in local optima.

$$AverageBest_{fit} = \frac{\sum Best_{fit}}{30} \tag{13}$$

All the methods are fed with a standard parameter suggested by the literature. The parameters used by the methods are listed in detail in Table 2.

VI. RESULTS

The performance and quality of the results of the proposed hybrid method (HSSOGA) are compared to its conventional GA and SSO methods in terms of mean (μ), standard deviation (σ), best fitness (optimal value), and average best fitness over 30 independent runs.

A. COMPARISON WITH CONVENTIONAL METHODS

The results are depicted in Tables 3 and 4, where the best results are shown in bold text. To ensure the convergence of the results, the method is processed 30 independent times on all the benchmark functions.

From the obtained results, we can see that HSSOGA has the best optimal value for 8 out of 11 test function problems, SSO has the best fitness on 5 out of 11 test function problems, and GA has the best optimal value on 4 out of 11 test function problems. On the other hand, the average best fitness value of 30 independent runs shows that HSSOGA is the best in 7 out of 11 test function problems, SSO is the best in 4 out of 11 test function problems, and GA is the best in 3 out of 11 test function problems. HSSOGA and SSO managed to obtain the global optimum value for functions 6 and 8, and it did not fall into local optimum, as the average is the same with the best fitness obtained. HSSOGA also obtained the best optimal value for function 7, as the best fitness and average best fitness are equal, which shows that HSSOGA's ability of exploration and exploitation outperforms both SSO and GA. From the results, we can conclude that HSSOGA outperformed GA and SSO in 2 unimodal and 6 multimodal functions (3, 4, 6, 7, 8, 9, and 11) in terms of obtaining the best fitness.

In terms of convergence, HSSOGA has faster convergence in functions 1, 2, 3, 4, 5, 6, 7, 9, and 11, while GA converges faster in function 10, and SSO converges faster in function 8, as depicted in Figure 3 (a-k), with mean values in Table 4. From the variance results shown in Table 4, it can be seen that HSSOGA is stable in functions 1, 2, 3, 4, 5, 6, 7, and 11, as they have a smaller dispersion between the values, while SSO has a small dispersion in values for functions 8, 9, and 10. However, SSO faces smaller dispersion because it falls into local optimum in functions 8 and 9, which are deemed to be unimodal functions.

To ensure the significance of the results obtained a One-way ANOVA with Post Hoc Tukey's test were carried out as depicted in Table 5.

From the statistical analysis we can say that HSSOGA outperforms GA in solving both unimodal and multimodal test functions. Besides, HSSOGA has a significant performance improvement towards GA in high ranges multimodal search space such as functions 6 and 9 as the p-Value and mean differences of HSSOGA and the conventional methods is significant at the 0.05 level.

B. COMPARISON WITH EXISTING METHODS

Based on the results from Tables 6 and 7, HSSOGA has the best fitness for 9 out of 11 test function problems and has the optimal fitness value in 5 out of 11 test function problems. HFPSO, SAGA, PSOGWO, and HSSOGSA, each have the best fitness for 2 out of 11 test function problems. By looking at the average best values after 30 independent runs, it can be concluded that HSSOGA is the best in 8 test functions, HPSOGA is the best in 4 test functions, and HFPSO and HSSOGA are the best in 1 of the 11 test functions. On the other hand, SAGA and PSOGWO did not have a good average best value in all of the test function problems. From the mean results, it can be said that HSSOGA outperformed HFPSO, HPSOGA, SAGA, PSOGWO, and HSSOGSA by obtaining the best average fitness values over 1000 iterations for the test functions 1, 2, 3, 4, 5, 6, 7, and 8. However, HPSOGA outperformed all the compared hybrid methods in test functions 9 and 10, while HPSOGA, SAGA, and HSSOGA had the same average fitness value for function 11. As such, it can be concluded that HSSOGA performs better than all the existing methods in all 5 unimodal functions and 4 multimodal functions.

From Figure 4 (a-k), it can be seen that HSSOGA has a faster convergence rate in functions 1, 2, 3, 4, 5, 6, 7, 8, and 11, while HPSOGA converges faster in functions 9 and 10. Moreover, HSSOGA has a smaller dispersion in values on functions 4, 7, and 8, while HPSOGA has a smaller dispersion in values on functions 1, 2, 3, 5, and 6. These dispersion values show the stability of the method on the functions stated above.

HSSOGA significantly outperformed many existing hybrid methods in terms of achieving the significant level of 0.05 in One-way ANOVA with Post Hoc Tukey's test as depicted in table 8. We can learn that the proposed method tends to have significance performance in solving majority of multimodal test functions such as function 6, 7, 8, 9 and 10 which proves that it has a good capability of avoiding local optima compared to other existing hybrid methods.

C. COMPARISON OF EXECUTION RUNTIMES

The average execution runtimes of the methods in eleven test functions are compared in Table 9. The average execution runtimes, R_{ave} over 30 independent runs over 1000 iterations each on eleven test functions are calculated to ensure the performance of the methods using the equation Eq. (14).

$$R_{ave} = \frac{\sum(R/30)}{11} \tag{14}$$

From the table we can see that the proposed HSSOGA takes slightly longer average execution runtime compared to its conventional methods. However, compared to existing memetic method (HPSOGA), the proposed method has a shorter execution runtime. This shows that the selection of SSO for the local search enables faster runtimes compared to the well-known PSO. The longer execution over 1000 iteration is caused by the crossover and mutation operators of GA to find the local optimum efficiently. HSSOGA is still said to be efficient because it converges towards the local optimum faster based on Figure 3 and 4 which directly reduce the average execution runtime to achieve global optimum in most test functions.

TABLE 7. The statistical comparison results with existing hybrid methods.

Test Function	HFPSO		HPSOGA		SAGA		PSOGWO		HSSOGSA		HSSOGA	
	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
F1	316.8884	2118.202	82.39804	**961.6036**	280.9659	2555.935	309.3164	2724.529	496.3401	3843.228	**72.48017**	1195.676
F2	3787.125	27152.86	1204.006	**14691.19**	4237.507	38202.84	3091.436	34630.06	5993.163	50129.45	**1197.833**	20002.23
F3	20849.44	392909.9	6.718748	**26.1917**	187.3455	4528.75	104616.5	3302224	4516.313	142115.6	**4.749502**	34.06526
F4	209665.8	3462915	114522.8	2751861	555880.4	6719246	298773.4	4466997	1103406	11063074	**107015.4**	**2621449**
F5	309.3069	1978.174	97.06827	**1114.291**	280.9695	2526.345	169.2135	2041.5	333.0854	3102.99	**70.72267**	1171.976
F6	2.820574	16.23711	0.977843	**10.95433**	3.530839	27.2045	4.775141	28.01791	3.906253	31.02391	**0.733629**	11.44752
F7	1.782764	2.910557	0.404207	1.75225	0.645342	2.644395	0.688493	2.729284	0.713476	2.468824	**0.167571**	**1.341744**
F8	45.74981	44.22613	11.74978	34.23369	18.1789	45.06898	55.54386	85.72036	77.20287	54.18906	**5.663088**	**29.45629**
F9	4206.01	932.7171	**1395.08**	1077.531	1661.989	1559.321	6609.242	1826.896	2923.363	**758.1194**	1474.353	1439.745
F10	-16.1301	5.230045	**-25.7479**	4.235901	-25.2285	4.229054	-15.8209	**2.548888**	-14.7504	3.436212	-25.2374	4.416332
F11	-45.2964	0.021993	**-45.2967**	0.031072	**-45.2967**	**0.014717**	-45.2768	0.248754	-45.2915	0.090301	**-45.2967**	0.017091

TABLE 8. Statistical analysis of results using "One-way ANOVA (Tukey's test)" between HSSOGA and the existing methods.

Test Function	Algorithm (I)	Algorithm (J)	Mean Difference (I-J)	p-Value (Sig.)
F1	HSSOGA	HFPSO	-244.40819	0.218
		HPSOGA	-9.91786	1.000
		PSOGWO	-236.83622	0.250
		SAGA	-208.48569	0.393
		HSSOGSA	**-423.85994***	**0.001***
F2	HSSOGA	HFPSO	-2589.29187	0.495
		HPSOGA	-6.17303	1.000
		PSOGWO	-1893.60315	0.794
		SAGA	-3039.67396	0.308
		HSSOGSA	**-4795.3297***	**0.015***
F3	HSSOGA	HFPSO	-20844.69	0.999
		HPSOGA	-1.97	1.000
		PSOGWO	-104611.78	0.518
		SAGA	-182.60	1.000
		HSSOGSA	-4511.56	1.000
F4	HSSOGA	HFPSO	-102650.39	0.999
		HPSOGA	-7507.39	1.000
		PSOGWO	-191758.02	0.980
		SAGA	-448864.98	0.545
		HSSOGSA	**-9223.37***	**0.003***
F5	HSSOGA	HFPSO	-238.58424	0.116
		HPSOGA	-26.34559	1.000
		PSOGWO	-98.49086	0.903
		SAGA	-210.24683	0.225
		HSSOGSA	-262.36277	0.061
F6	HSSOGA	HFPSO	-2.08694	0.295
		HPSOGA	-0.24421	1.000
		PSOGWO	**-4.04151***	**0.001***
		SAGA	-2.79721	0.058
		HSSOGSA	**-3.17262***	**0.019***
F7	HSSOGA	HFPSO	**-1.61519***	**<0.001***
		HPSOGA	-0.23664	0.225
		PSOGWO	**-0.52092***	**<0.001***
		SAGA	**-0.47777***	**<0.001***
		HSSOGSA	**-0.54590***	**<0.001***
F8	HSSOGA	HFPSO	**-40.08672***	**<0.001***
		HPSOGA	-6.08669	0.095
		PSOGWO	**-49.88077***	**<0.001***
		SAGA	**-12.51581***	**<0.001***
		HSSOGSA	**-71.53978***	**<0.001***
F9	HSSOGA	HFPSO	**-2731.6565***	**<0.001***
		HPSOGA	79.27356	0.761
		PSOGWO	**-5134.8882***	**<0.001***
		SAGA	**-187.63607***	**0.019***
		HSSOGSA	**-1449.0094***	**<0.001***
F10	HSSOGA	HFPSO	**-9.10732***	**<0.001***
		HPSOGA	0.51047	0.060
		PSOGWO	**-9.41652***	**<0.001***
		SAGA	-0.00891	1.000
		HSSOGSA	**-10.48700***	**<0.001***
F11	HSSOGA	HFPSO	-0.00027	1.000
		HPSOGA	-0.00074	1.000
		PSOGWO	**-0.01992***	**0.001***
		SAGA	-0.00005	1.000
		HSSOGSA	-0.00528	0.891

***. The mean difference is significant at the 0.05 level.**

D. OVERALL RESULTS SUMMARY

The overall results are summarized by ranking all the methods in terms of the mean fitness value obtained. The ranking summary is presented in Table 10. From Table 10, we can conclude that HSSOGA outperformed all the other compared methods in 8 out of 11 test function problems, where it shows a good quality of balance between exploration and exploitation in a method. However, in functions 7 and 8, where these functions are highly multimodal, HSSOGA is ranked second. This is because the method possesses a high velocity of the SSO, which makes the method miss the global optimum in earlier stages. For function 9, the test function has valleys that make it difficult to search. As such, GA, which has a good exploration characteristic, manages to find the best fitness efficiently, whereases HSSOGA is ranked third, as merging fast SSO compromises the exploration of GA slightly.

TABLE 9. The average execution runtimes of the proposed, conventional, and exiting methods.

Algorithms	Average execution runtimes over 11 test functions (seconds)
HSSOGA	3.698654
GA	1.870322
SSO	1.546674
HFPSO	0.706595
HPSOGA	3.738738
PSOGWO	0.486333
SAGA	2.193528
HSSOGSA	8.465875

VII. DISCUSSION

A. SPACE AND TIME COMPLEXITY

Space complexity is the amount of space required throughout the execution of the algorithm. In the proposed HSSOGA method there are three arrays that holds the population which are the initial population, crossover population and mutated population. The initial population is an array with the length of n. The crossover population has a length of $pc * n$ where pc is the crossover probability and mutated population has the length of pm*n where pm is the crossover probability. These populations are merged finally before truncated to the length of n to find the best solutions. This makes the overall space complexity as such $(O(n) + O(pc * n) + O(pm * n)) * O(n) = O(n^2)$. Since the method does not tune the crossover and mutation rates over the iterations it only requires less space throughout the execution as the crossover and mutation rates are set to 0.7 and 0.3. The space complexity of SSO is relatively lower compared to HSSOGA as it does not have multiple population and merging of populations. SSO needs space to store the personal best value to drive the method towards reaching the global best solution which makes the space complexity of SSO, $O(n)$. On the other hand, GA's space complexity is almost the same as HSSOGA's space complexity because it has several populations that are later merged and truncated which makes its space complexity $O(n^2)$. Time complexity is the amount of time it takes generally for the algorithm to run. We use population size, n and it is looped until the maximum criteria reached which makes the complexity $O(n)$. The complexity to calculate the functions is $O(1)$. During the process of execution, the population from crossover and mutations are merged and sorted which takes a complexity of $O(nlogn)$. So, the overall complexity can be calculated as such $O(1) + O(n) + O(nlogn) = O(nlogn)$. The time complexity of SSO is lower than GA and HSSOGA because SSO does not have merging or sorting operators that makes the time complexity of SSO to be $O(1)+O(n) = O(n)$. Since, GA is known for its merging and sorting of crossover and mutation populations, it has a time complexity similar to HSSOGA of $O(1)+O(n)+O(nlogn) = O(nlogn)$. However, HSSOGA might take longer runtime, R to run the algorithm compared to SSO and GA, but the time taken to obtain global solution will be faster compared to the other methods because of its balance between exploration and exploitation capabilities.

B. OVERALL DISCUSSION SUMMARY

GA is well known for its exploration capability, but it has a slower convergence rate because it limits exploitation. On the other hand, SSO is known as a swarm-based method that has strong exploitation capability. GA's slow convergence avoids SSO from falling into local optima, and SSO's precision enables finding of the true optimal value. These instances motivated us to merge both GA and SSO, forming a hybrid method called HSSOGA. The combination of the functionalities of these methods helps the algorithm to explore and exploit a search space domain efficiently with a faster convergence rate, which in turn helps it in obtaining the global optimal values.

Eleven well-known test function problems are used to evaluate the performance of the proposed method, where 5 of the functions are unimodal functions and 6 are multimodal functions. Multimodal functions are a good test for the hybrid algorithms as they ensure that the method has a good exploration and exploitation capability.

On comparing with the conventional method, it is seen that HSSOGA performs well in all the multi-modal problems except for test function 10 where GA, which has exploration capability, performs better. From the comparison with existing hybrid methods, it was observed that HSSOGA outperformed them in both unimodal and multimodal problems. However, HPSOGA outperformed the proposed method in functions 10 and 11. HSSOGA also shows that it can perform well in large scale optimization problems, as it outperforms all the compared methods in function 6, which has a large search domain. In a nutshell, HSSOGA can perform well in wide search space domains with multiple local optima as compared to the conventional and existing hybrid methods, as it has a good balance between the exploration and exploitation capability.

The advantages of HSSOGA are outlined below:

- HSSOGA has a lower time complexity as it adapts to LTH methods where both algorithms are run in parallel.
- Adjusting GA parameters such as crossover percentage (pc), mutation percentage (pm), and mutation rate (mu) will enhance the global search capability, while adjusting SSO parameters such as pH value and temperature will enhance the local search capability.
- HSSOGA stores the x_{sgBest} value in its memory where each candidate solution (sperm) can observe and gravitate towards it at any time.

Evaluating the proposed method in test functions ensures the performance of the method in a standard defined problem; however, the performance of the methods can vary in real situation implementations which have different parameters. To ensure that HSSOGA performs efficiently in all situations and implementations, it can be evaluated in solving real-life problems in future. Tweaking the parameters of the proposed method may improve the performance of the method in different situations and test functions. As such, we will apply the proposed method (HSSOGA) to have adaptive and dynamic parameters for better efficiency. The proposed

TABLE 10. Methods ranking based on the statistical results.

Test Function	SSO	GA	HFPSO	HPSOGA	SAGA	PSOGWO	HSSOGSA	HSSOGA
F1	3	7	6	2	4	5	8	**1**
F2	3	7	5	2	6	4	8	**1**
F3	4	6	8	2	3	7	5	**1**
F4	5	8	4	3	7	6	2	**1**
F5	3	7	6	2	5	4	8	**1**
F6	3	5	4	2	6	8	7	**1**
F7	2	4	8	3	5	6	7	**1**
F8	**1**	4	6	3	5	7	8	2
F9	8	3	6	**1**	4	7	5	2
F10	8	**1**	5	2	4	6	7	3
F11	6	4	4	**1**	**1**	8	7	**1**

TABLE 11. Unimodal and multimodal test function problems used in this study.

Test Function	Func. Name	Formula	Dim	Range	Optima
F1	Sphere	$f_1(x) = \sum_{i=1}^{d} x_i^2$	30	[-100, 100]	0
F2	Sum Square	$f_2(x) = \sum_{i=1}^{d} ix_i^2$	30	[-100, 100]	0
F3	Zakharov	$f_3(x) = \sum_{i=1}^{d} x_i^2 + (\sum_{i=1}^{d} 0.5ix_i^2)^2 + (\sum_{i=1}^{d} 0.5ix_i^2)^4$	30	[-5, 10]	0
F4	Rosenbrock	$f_4(x) = \sum_{i=1}^{d-1}[100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	30	[-30, 30]	0
F5	Step	$f_5(x) = \sum_{i=1}^{d}(x_i + 0.5)^2$	30	[-100, 100]	0
F6	Griewank	$f_6(x) = \sum_{i=1}^{d} \frac{x_i^2}{4000} - \prod_{i=1}^{d} \cos \frac{x_i}{\sqrt{i}} + 1$	30	[-600, 600]	0
F7	Ackley	$f_7(x) = -a \exp(-b\sqrt{\frac{1}{d} \sum_{i=1}^{d} x_i^2}) - \exp(\frac{1}{d} \sum_{i=1}^{d} \cos(cx_i)) + a + \exp(1)$	30	[-32, 32]	0
F8	Rastrigin	$f_8(x) = 10d + \sum_{i=1}^{d}[x_i^2 - 10\cos(2\pi x_i)]$	30	[-5.12, 5.12]	0
F9	Schwefel 2.26	$f_9(x) = 418.9829d - \sum_{i=1}^{d} x_i \sin(\sqrt{\lvert x_i \rvert})$	30	[-500, 500]	0
F10	Michalewicz	$f_{10}(x) = \sum_{i=1}^{d} \sin(x_i) \sin^{2m}(\frac{ix_i^2}{\pi})$	30	[-π, π]	<-9.66015
F11	EggCrate	$f_{11}(x) = x_1^2 + x_2^2 + 25(\sin^2 x_1 + \cos^2 x_2)$	30	[-5, 5]	0

method will also be implemented to solve real-life issues such as those in energy-efficient clustered wireless sensor networks (WSNs) [3], parameter estimation for frequency-modulated sound waves [41], the leather nesting problem (LNP) [42], and the multimodal home-healthcare scheduling (MHS) problem [4].

VIII. CONCLUSION AND FUTURE WORKS

In this paper, a new hybrid sperm swarm optimization and genetic algorithm, HSSOGA, is proposed, that merges the exploitation capability of SSO and the exploration capability of GA. The proposed method uses GA's mutation and roulette wheel selection operators to explore the search space, which helps the SSO to escape from the local minima of multimodal functions. A standard test consisting of 11 test problems, of which 5 are unimodal and 6 are multimodal, was used for evaluations, to ensure that the proposed method will be able to give a good balance between the exploration and exploitation capability. To evaluate the performance, HSSOGA is compared to conventional GA and SSO, as well as some existing hybrid methods such as HFPSO, SAGA, PSOGWO, HPSOGA, and HSSOGSA. The results are represented in both quantitative and qualitative approaches. The best fitness value, mean, and standard deviation are categorized as quantitative approaches, while the comparison of convergence rate is referred to as a qualitative approach. From the results obtained from Tables 2, 3, 4 and 5, it was observed that HSSOGA outperformed the conventional methods by achieving better values near to global optimum in functions 3, 4, 5, 6,7, 8, 9, and 11, and having a faster convergence rate in functions 1, 2, 3, 4, 5, 6, 7, 9, and 11, as illustrated in the figures above. Besides, HSSOGA also outperformed the existing hybrid method in converging towards a better global optimum value in functions 1, 2, 3, 5, 6, 7, 8, 9, and 11. It also converged faster in functions 1, 2, 3, 4, 5, 6, 7, 8, and 11, as depicted in figures above.

In conclusion, it can be inferred that HSSOGA can avoid being trapped in local optima and achieve the global optima values efficiently, which are essential in solving real-life problems. Therefore, we recommend using the proposed method in the field of engineering, healthcare, and networking. In future, we hope this work can be continued with more development such as dynamic and adaptive parameter tuning, as well as the implementation of large-scale global optimization problems. We also would also test the proposed method with clustering in wireless sensor networks.

APENDIX A

The 11-test function problem formula with its features according to [34] are presented in Table 11.

REFERENCES

[1] N. Xiong, D. Molina, M. L. Ortiz, and F. Herrera, "A walk into metaheuristics for engineering optimization: Principles, methods and recent trends," *Int. J. Comput. Intell. Syst.*, vol. 8, no. 4, pp. 606–636, 2015.

[2] X.-S. Yang, "Optimization and metaheuristic algorithms in engineering," in *Metaheuristics in Water, Geotechnical and Transport Engineering*, X.-S. Yang et al. Oxford, U.K.: Elsevier, 2013, ch. 1, pp. 1–23.

[3] D. R. Prasad, P. V. Naganjaneyulu, and K. S. Prasad, "Metaheuristic techniques for cluster selection in WSN," in *Proc. Int. Conf. Algorithms, Methodol., Models Appl. Emerg. Technol. (ICAMMAET)*, Feb. 2017, pp. 1–6.

[4] G. Hiermann, "Metaheuristics for solving a multimodal home-healthcare scheduling problem," *Central Eur. J. Oper. Res.*, vol. 23, pp. 89–113, Mar. 2015.

[5] C.-W. Tsai, M.-C. Chiang, A. Ksentini, and M. Chen, "Metaheuristic algorithms for healthcare: Open issues and challenges," *Comput. Electr. Eng.*, vol. 53, pp. 421–434, Jul. 2016.

[6] X.-S. Yang, S. Deb, and S. Fong, "Metaheuristic algorithms: Optimal balance of intensification and diversification," *Appl. Math. Inf. Sci.*, vol. 8, no. 3, pp. 977–983, May 2014.

[7] L. Cao, Y. Cai, and Y. Yue, "Swarm intelligence-based performance optimization for mobile wireless sensor networks: Survey, challenges, and future directions," *IEEE Access*, vol. 7, pp. 161524–161553, 2019.

[8] J. Xu and J. Zhang, "Exploration-exploitation tradeoffs in metaheuristics: Survey and analysis," in *Proc. 33rd Chin. Control Conf.*, 2014, pp. 8633–8638.

[9] L. Bianchi, M. Dorigo, L. M. Gambardella, and W. J. Gutjahr, "A survey on metaheuristics for stochastic combinatorial optimization," *Natural Comput.*, vol. 8, no. 2, pp. 239–287, 2019.

[10] F. Fausto, A. Reyna-Orta, E. Cuevas, Á. G. Andrade, and M. Perez-Cisneros, "From ants to whales: Metaheuristics for all tastes," *Artif. Intell. Rev.*, vol. 53, no. 1, pp. 753–810, Jan. 2020.

[11] M. Abdel-Basset, L. Abdel-Fatah, and A. K. Sangaiah, "Chapter 10—Metaheuristic algorithms: A comprehensive review," in *Computational Intelligence for Multimedia Big Data on the Cloud with Engineering Applications*, A. K. Sangaiah, M. Sheng, and Z. Zhang, Eds. Cambridge, MA, USA: Academic Press, 2018, pp. 185–231.

[12] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. Int. Conf. Neural Netw.*, 1995, pp. 1942–1948.

[13] H. A. Shehadeh, I. Ahmedy, and M. Y. I. Idris, "Sperm swarm optimization algorithm for optimizing wireless sensor network challenges," in *Proc. 6th Int. Conf. Commun. Broadband Netw. (ICCBN)*, Singapore: Association for Computing Machinery, 2018, pp. 53–59.

[14] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014.

[15] R. Storn and K. Price, "Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optim.*, vol. 11, no. 4, pp. 341–359, 1997.

[16] J. H. Holland, "Genetic algorithms," *Sci. Amer.*, vol. 267, no. 1, pp. 66–72, Jul. 1992.

[17] N. M. Sabri, M. Puteh, and M. R. Mahmood, "An overview of gravitational search algorithm utilization in optimization problems," in *Proc. IEEE 3rd Int. Conf. Syst. Eng. Technol.*, Aug. 2013, pp. 61–66.

[18] D. Bertsimas and J. Tsitsiklis, "Simulated annealing," *Statist. Sci.*, vol. 8, no. 1, pp. 10–15, 1993.

[19] A. Askarzadeh and E. Rashedi, "Harmony search algorithm: Basic concepts and engineering applications," in *Recent Developments in Intelligent Nature-Inspired Computing*, S. Patnaik, Ed. Hershey, PA, USA: IGI Global, 2017, pp. 1–36.

[20] S. Garshasbi, Y. Mohammadi, S. Graf, S. Garshasbi, and J. Shen, "Optimal learning group formation: A multi-objective heuristic search strategy for enhancing inter-group homogeneity and intra-group heterogeneity," *Exp. Syst. Appl.*, vol. 118, pp. 506–521, Mar. 2019.

[21] I. B. Aydilek, "A hybrid firefly and particle swarm optimization algorithm for computationally expensive numerical problems," *Appl. Soft Comput.*, vol. 66, pp. 232–249, May 2018.

[22] A. Y. Prasad and B. Rayanki, "A generic algorithmic protocol approaches to improve network life time and energy efficient using combined genetic algorithm with simulated annealing in MANET," *Int. J. Intell. Unmanned Syst.*, vol. 8, no. 1, pp. 23–42, Jul. 2019.

[23] F. A. Şenel, F. Gökçe, A. S. Yüksel, and T. Yiğit, "A novel hybrid PSO–GWO algorithm for optimization problems," *Eng. Comput.*, vol. 35, no. 4, pp. 1359–1373, Oct. 2019.

[24] H. Garg, "A hybrid PSO-GA algorithm for constrained optimization problems," *Appl. Math. Comput.*, vol. 274, pp. 292–305, Feb. 2016.

[25] H. A. Shehadeh, "A hybrid sperm swarm optimization and gravitational search algorithm (HSSOGSA) for global optimization," *Neural Comput. Appl.*, vol. 33, no. 18, pp. 11739–11752, Sep. 2021.

[26] V. K. Chawra and G. P. Gupta, "Load balanced node clustering scheme using improved memetic algorithm based meta-heuristic technique for wireless sensor network," *Proc. Comput. Sci.*, vol. 167, pp. 468–476, Jan. 2020.

[27] C. Ryan, "Evolutionary algorithms and metaheuristics," in *Encyclopedia of Physical Science and Technology*, 3rd ed., R. A. Meyers, Ed. New York, NY, USA: Academic Press, 2003, pp. 673–685.

[28] G. Poonam, "A comparison between memetic algorithm and genetic algorithm for the cryptanalysis of simplified data encryption standard algorithm," *Int. J. Netw. Secur. Appl.*, vol. 1, no. 1, pp. 34–42, 2009.

[29] H. A. Shehadeh, H. M. Mustafa, and M. Tubishat, "A hybrid genetic algorithm and sperm swarm optimization (HGASSO) for multimodal functions," *Int. J. Appl. Metaheuristic Comput.*, vol. 13, no. 1, pp. 1–33, 2022.

[30] N. El-Omari, "Sea lion optimization algorithm for solving the maximum flow problem," *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 5, pp. 388–395, 2021.

[31] H. A. Shehadeh, I. Ahmedy, and I. M. Y. Idris, "Empirical study of sperm swarm optimization algorithm," in *Intelligent Systems and Applications*. Cham, Switzerland: Springer, 2019.

[32] P. Bajpai and M. Kumar, "Genetic algorithm—An approach to solve global optimization problems," *Indian J. Comput. Sci. Eng.*, vol. 1, pp. 199–206, Oct. 2010.

[33] N. Razali and J. Geraghty, "Genetic algorithm performance with different selection strategies in solving TSP," in *Proc. World Congr. Eng.*, vol. 2, 2011, pp. 1–6.

[34] S. Surjanovic and D. Bingham. (2013). *Virtual Library of Simulation Experiments: Test Functions and Datasets*. [Online]. Available: https://www.sfu.ca/ ssurjano/index.html

[35] L. Jourdan, M. Basseur, and E.-G. Talbi, "Hybridizing exact methods and metaheuristics: A taxonomy," *Eur. J. Oper. Res.*, vol. 199, no. 3, pp. 620–629, Dec. 2009.

[36] A. Norouzi and A. Zaim, "Genetic algorithm application in optimization of wireless sensor networks," *Sci. World J.*, vol. 2014, Feb. 2014, Art. no. 286575.

[37] A. George, B. R. Rajakumar, and D. Binu, "Genetic algorithm based airlines booking terminal open/close decision system," in *Proc. Int. Conf. Adv. Comput., Commun. Informat. (ICACCI)*, Chennai, India: Association for Computing Machinery, 2012, pp. 174–179.

[38] J. Zhang, H. S. Chung, and W. L. Lo, "Pseudocoevolutionary genetic algorithms for power electronic circuits optimization," *IEEE Trans. Syst., Man, Cybern., C* , vol. 36, no. 4, pp. 590–598, Jul. 2006.

[39] S. Katoch, S. S. Chauhan, and V. Kumar, "A review on genetic algorithm: Past, present, and future," *Multimedia Tools Appl.*, vol. 80, no. 5, pp. 8091–8126, Oct. 2021.

[40] J. D. Neves and M. F. Bahia, "Gels as vaginal drug delivery systems," *Int. J. Pharmaceutics*, vol. 318, nos. 1–2, pp. 1–14, Aug. 2006.

[41] B. Gothania, G. Mathur, and R. Yadav, "Accelerated artificial bee colony algorithm for parameter estimation of frequency-modulated sound waves," *Int. J. Electron. Commun. Eng.*, vol. 7, no. 1, pp. 63–74, 2014.

[42] A. Crispin, P. Clay, G. Taylor, T. Bayes, and D. Reedman, "Genetic algorithm coding methods for leather nesting," *Int. J. Speech Technol.*, vol. 23, no. 1, pp. 9–20, Jul. 2005.

[43] E. G. Talbi, "A taxonomy of hybrid metaheuristics," *J. Heuristics*, vol. 8, no. 5, pp. 541–564, Sep. 2002.

[44] Y. Sharafi, M. A. Khanesar, and M. Teshnehlab, "COOA: Competitive optimization algorithm," *Swarm Evolut. Comput.*, vol. 30, pp. 39–63, Oct. 2016.

[45] M. Abd Elaziz, A. A. Ewees, N. Neggaz, R. A. Ibrahim, M. A. A. Al-Qaness, and S. Lu, "Cooperative meta-heuristic algorithms for global optimization problems," *Exp. Syst. Appl.*, vol. 176, Aug. 2021, Art. no. 114788.

[46] A. H. Raed, S. N. Shahab, and M. A. Abdullah, "Correlation with the fundamental PSO and PSO modifications to be hybrid swarm optimization," *Iraqi J. Comput. Sci. Math.*, vol. 2, no. 2, pp. 25–32, 2021.

BRYAN RAJ was born in Ipoh, Perak, Malaysia, in 1997. He received the B.CompSc. degree in computer systems and networking from the University Malaya, Malaysia, in 2020, where he is currently pursuing the Ph.D. degree in computer science.

Throughout his bachelor's degree program, from the year 2016 to 2020, he was a Laboratory Demonstrator for basic networking class with the Universiti Malaya. He is also working as a Graduate Research Assistant at the Universiti Malaya. His research interests include wireless sensor networks, clustering algorithms, and methods and optimization algorithms. He was a recipient of the Dean's List Award for all seven semesters in his bachelor's degree.

ISMAIL AHMEDY (Member, IEEE) received the B.Sc. degree in computer science from the Universiti Teknologi Malaysia, in 2006, the M.Sc. and Ph.D. degrees in computer science from The University of Queensland, Australia, in 2009 and 2015, respectively, and the Ph.D. degree from the Universiti Teknologi Malaysia specializing in communication protocol for wireless sensor networks.

He is currently a Senior Lecturer at the Department of Computer System and Technology, Faculty of Computer Science and Information Technology, Universiti Malaya. He is also a Huawei Certified Academy Instructor and a Network Associate. He is supervising around ten postgraduate students currently and published more than 15 articles to ISI indexed journals in the year 2021 and 2020 alone. His research interests include wireless sensor networks, the Internet of Things, energy management, and optimization algorithm. After completing his study, he has been granted a full scholarship to pursue his studies in master's and Ph.D. degrees. He also received the Best Online Presentation Award from the 3rd International Conference on Information Science and Systems (ICISS) in 2020.

MOHD YAMANI IDNA IDRIS (Member, IEEE) received the bachelor's degree in electrical engineering, the M.CompSc. degree in computer science, and the Ph.D. degree in system on chip from the Universiti Malaya, Malaysia.

He is currently an Associate Professor with the Department of Computer System and Technology, Faculty of Computer Science and Information Technology, Universiti Malaya. He is supervising more than ten post graduate student and have published more than 15 articles to ISI indexed journals in the year 2021 and 2020 alone. His research interests include robotics, system on chip, and image and signal processing. He also have received the Excellence Service Award from Universiti Malaya, in 2020.

RAFIDAH MD. NOOR received the bachelor's degree in IT from the University Utara Malaysia, in 1998, the M.Sc. degree in computer science from the Universiti Teknologi Malaysia (UTM), in 2000, and the Ph.D. degree in computing from Lancaster University, U.K., in 2010.

She is currently a Professor with the Department of Computer System and Technology, Faculty of Computer Science and Information Technology, Universiti Malaya, and the Director of the Centre of Mobile Cloud Computing Research, which focuses on high-impact research. She has performed more than two million (Ringgit Malaysia) research grants from internal and external entities. She is supervising more than 20 post graduate student and have published more than 15 articles to ISI indexed journals in the year 2021 and 2020 alone. Her research interests include transportation, mobile network technologies, and mobile communication devices.