BIS: A New Swarm-Based Optimisation Algorithm

Fevzi Tugrul Varna
Department of Informatics, University of Sussex
Brighton, United Kingdom
f.varna@sussex.ac.uk

Phil Husbands
Department of Informatics, University of Sussex
Brighton, United Kingdom
philh@sussex.ac.uk

***Abstract*—This paper presents a novel swarm-based search algorithm: the bio-breeding intelligent swarm (BIS) algorithm. BIS agents imitate the offspring and maturity phases of the typical lifecycle of an animal. As in nature, the BIS algorithm makes gender distinction among agents and the main search strategy exploits competition between male agents in an attempt to provide a better location for females. BIS agents embark on various nature-inspired mating strategies and the inspiration for the reproduction model is derived from temperature-dependent sex determination (TSD), a reptilian reproduction system. The BIS algorithm's TSD inspired reproduction model enables female agents to control the gender of offsprings based on guidance provided by their male mates, subsequently resulting in regulation of the male-female ratio in the swarm which in turn auto-controls the balance of exploration and exploitation within the population of agents. The efficiency of the BIS algorithm was tested over a wide range of benchmarks including unconstrained high dimensional and real-world problems. The BIS algorithm performed very well in comparison with a number of leading population-based stochastic search methods, finding the highest number of global optimums.**

***Index Terms*—swarm intelligence, optimisation, metaheuristics**

I. Introduction

Swarm intelligence (SI) algorithms have become a popular and powerful class of stochastic search algorithms that have been shown to perform well across a range of problem types [1] [2] [3]. Much of their power, like most other metaheuristics, derives from underlying stochastic mechanisms that provide a good balance between exploration and exploitation. The most powerful characteristic of SI algorithms is that simple behaviour of individual agents culminates in complex, hard-to-predict and sophisticated group behaviour. Hence, SI algorithms are widely used for various real-world discrete [4] [5] [6] [7] and continuous [8] [9] [10] [11] problems, and there is a continuous effort to improve and refine them. The majority of SI algorithms employ forms of courtship, nest building, reproduction and/or various other biological inspirations as the core mechanism to control exploration/exploitation, and hence the search process. However, these strategies tend to be very simple, abstract or arbitrary. The general approach of most swarm-based algorithms is to adopt a central search behaviour for agents, derived from some inspiration. However, the study presented here aims to pursue a different approach by designing an entire lifecycle for artificial agents besides the central search behaviour. The process of designing the agent's lifecycle includes the modelling of an offspring phase and a maturity phase (detailed later), taking direct inspiration from the mating strategies and reproduction systems of animals. Gender distinction was an infrequently used mechanism in the early literature of metaheuristics, although more recent algorithms [12] [13] [14] tend to employ such mechanism to enhance the level of labour distribution among agents. The present study aims to design a more powerful gender distinction mechanism with auto-controlled adaptive gender density enforced by a bio-inspired reproduction system to enable an even more effective division of labour. By allowing the algorithm to auto-regulate gender density, the proposed mechanism also aims to auto-control the sub-populations of agents assigned for exploration and exploitation. Multiple social interactions are one of four concepts that comprise self-organisation [15] and in most swarm-based algorithms, it's commonly achieved through mechanisms such as explicit/implicit communication. The work presented here models highly stochastic multiple social-interaction mechanisms by combining the usage of different mating strategies, male-female attraction and dynamically-changing social groups. Certain animals such as hyenas and giraffes form dynamic groups or herds using hierarchies and the present study intends to model dynamically changing individual hierarchies/ranks based on certain rules to form social groups and these social groups are expected to provide nondeterministic suggestions for agents' movements. It has been suggested that one of the issues in relation to modern metaheuristics in general is that the majority of algorithms proposed are primarily tested using small-scale and artificial problems [16]. However nowadays it is common to encounter problems with a considerable number of dimensions. In fact, with the broadening of the metaheuristic literature, large-scale problems are receiving more attention from researchers. Therefore, in light of this, a major motivation for the present study is the development of a new swarm-based algorithm capable of solving large-scale and real-world problems in reasonable computation time. Many recent variants of metaheuristics, such as [17] [18], are observed to have much more complex structure than their canonical versions. These variants are discernibly more efficient and are able to handle more diverse problems. Hence the algorithm proposed in the present work is not unusual in comprising a more complex structure and set of interaction mechanisms in comparison to canonical algorithms. However, its individual elements are all simple. Indeed the slightly more

978-1-7281-2547-3/20/$31.00 ©2020 IEEE

complex framework of the proposed algorithm is inspired by the observation that biological degeneracy plays a vital role in boosting evolvability in nature, and therefore can improve the efficiency of search processes [19]. In addition, the introduced algorithm may be categorised as a hybrid quasi-PSO algorithm with behavioural inspirations from nature. The present work introduces an explicit idea of a lifecycle where the search behaviour of the agents changes throughout the search process. In each phase of the lifecycle, behaviour of agents acquire elements from existing algorithms in the literature [50] [51] [52]. As a result, the proposed algorithm provides a degenerate system which is very useful in terms of adaptation. Biological degeneracy – whereby multiple interacting mechanisms enable multiple different ways of producing an outcome - is an ubiquitous property of biological systems at all levels of organisation [20]. [21] reveals that systems with simple redundancy have considerably lower evolvability than degenerate (e.g. highly versatile) systems with selectable changes of behaviour which enable compensatory actions to occur within the system. According to [20], degeneracy is an essential part of natural selection and besides environmental impact is one of the main drivers for the evolution of complexity.

II. Proposed Algorithm:BIS

Foraging, courtship, and nest-building behaviours of various living beings have been a successful inspiration for numerous swarm-based algorithms. This paper proposes a novel algorithm inspired by a peculiar reproduction strategy of reptiles called temperature-dependent sex determination (TSD) and various mating strategies used by different animals. Members of the swarm carry out various search behaviours determined by their current status and role in the population. Together these combine to produce a powerful optimization algorithm. In the next section, several concepts used as inspirations are introduced to provide the reader with a better understanding of the proposed work.

A. Temperature-dependent sex determination

Most animals employ a system of genetic sex determination (GSD) [22] however, in many reptilian species, egg incubation temperature determines the gender of offspring. Eggs incubated at higher temperatures during gonad formation yields male offsprings and on the contrary, eggs incubated at lower temperatures develop as females. The pivotal temperature is the constant incubation temperature, or simply the threshold, and eggs incubating at the pivotal temperature produce an equal number of both sexes [23]. The algorithm proposed in this paper adopts a simple mechanism based on TSD to control gender determination.

B. Mating strategies

Reproduction is a fundamental feature of all life and the type of strategy adopted is determined by many factors with regards to social organisation, dispersal by gender and habitat types. Although, in sexual animals, the type of mating system varies, the most common mating strategies include promiscuity, polyandry, and polygyny [24]. Besides these predominant strategies, more specialised strategies such as resource defence polyandry, harems, cooperative polyandry and lekking exist [25]. The common type of polyandry refers to a single female associated with two or more males, it provides females with multiple mating options; hence it's advantageous for increasing the female's rate of reproductive success. Harem formation is observed in various bird species consists of a group of females associated with a single male. Lekking, on the contrary, is involves a group of males seeking female mates. Leks are not usually associated with any resource; instead, males may form groups nearby females' travelling paths to attract passing females. It is assumed that males prefer lekking instead of performing solitary courtship behaviour to increase their chances of mating and rate of reproductive success [26].

C. BIS agent's lifecycle

1) Offspring phase: Biologically, animals reproduce asexually or sexually. For some specifies, asexual reproduction is advantageous if the organism is properly adapted to their environments, but in case of environmental change, it may be disadvantageous for offsprings as identical genotypes inhibits change and dampens the evolutionary process. On the contrary, sexual reproduction uses genes from both parents, allowing the possibility of variation in the offspring resulting in the support of the evolutionary process. The offspring phase in the present work is designed with inspiration from the latter to enable agents to change and adapt. An offspring's learning is highly influenced by the father (fixed) and a mother (selected randomly from the available single females, depending on the mating strategy). An offspring's dependency on its parents continues until it reaches maturity, and the period of time agents spend as offsprings can be adjusted to control the phase. As mentioned previously, during this period the offspring is constrained from mate searching and reproducing. On reaching maturity, an agent starts learning according to its gender and mate status and follows through the cycle in the maturity phase (mate searching and reproduction).

2) Maturity phase: The maturity phase comprises the main phase of the agent's lifecycle and it constitutes the process of mate searching and reproduction. Competing behaviour is prevalently observed in animals, it commonly erupts as a result of more than one animal attempting to access a limited resource (e.g. food, nest, etc) or occasionally a mate. In the present work, the idea of competition, more specifically intraspecific competition, is employed among males. Single male agents compete to find mates, and males already with a mate compete in finding a better location for female mates to avoid abandonment. The location or guidance provided by the mated male is particularly important as it plays a significant role in determining the gender of offspring. Ultimately, all males compete to sustain the female population in the swarm to perpetuate reproduction for subsequent generations. During the maturity phase, a single male agent's primary focus is to perform a local search on the region of the current best solution, and mated male agents carry out a local search around their current position while simultaneously learning from their social group. Social grouping is employed as a mechanism to create and maintain a multi-interaction among agents and support stochastic learning throughout the search process. To form social groups, agents are assigned hierarchies based on certain qualities, and the given hierarchies are updated (if necessary) at each iteration to form dynamically-changing social groups. The following rules are used to assign a hierarchy to agents: (1) Agents that discover the best positions. (2) Mated male/females and all lekking agents. (3) non-lekking single male/female agents. Similar to males, females also exhibit distinct behaviour based on their mating status. Single females perform exploration and females with mates rely on positions provided by their male mates while also moving towards the best-known position. These mechanisms interact to produce a highly efficient search process.

a) Mate searching model: The mate searching model of the BIS algorithm is composed of two mechanisms, namely, mating strategy and mate selection, in addition to certain restrictions. Those restrictions require agents to be single and in the maturity phase. The mating strategy is inspired by real strategies acquired by animals and agents adopt a specific mating strategy based on the following rules: (1) Females adopt polyandry strategy upon discovery of the new best solution. (2) Females adopt the lekking strategy if their current position is improved (but not the best). (3) Single males adopt lekking strategy for a specified period (specified by the ψ parameter). (4) Males adopt harem formation upon discovery of a new best solution. On the other hand, the mate selection mechanism enforces the selector agent (could be male or female depending on the strategy) to choose the most dissimilar available agent of the opposite sex. This selection mechanism aims to boost diversity of offspring in case reproduction occurs.

b) Reproduction model: The reproduction model of the BIS algorithm is based on several rules. Reproduction is only permitted for mature agents. Following mate selection, male and female agents must remain bound for a reasonable amount of time to be able to reproduce, and the reproduction frequency parameter determines this time. When mated, female agents rely on their mates to guide them to a better position (further discussed later). The quality of solution provided by the male has an impact on two states, anti-aphrodisiac and temperature, hence providing quality guidance contributes to maintaining their pair bond, the level of anti-aphrodisiac and temperature. Male agents use anti-aphrodisiac, applied to their female mates, as a form of repelling strategy to deflect competing suitors; satisfactory guidance from males increase anti-aphrodisiac (it decreases otherwise) resulting in longer periods of bonding between pairs. Regulation of the level of anti-aphrodisiac and temperature are carried out at specific intervals determined by the δ parameter. In cases where anti-aphrodisiac decreases below a certain level, the female abandons the male mate hence reproduction does not take place. On the contrary, if the bond is maintained for the time required by the reproduction frequency parameter, offspring are produced, and their gender is determined by the TSD reproduction mechanism.

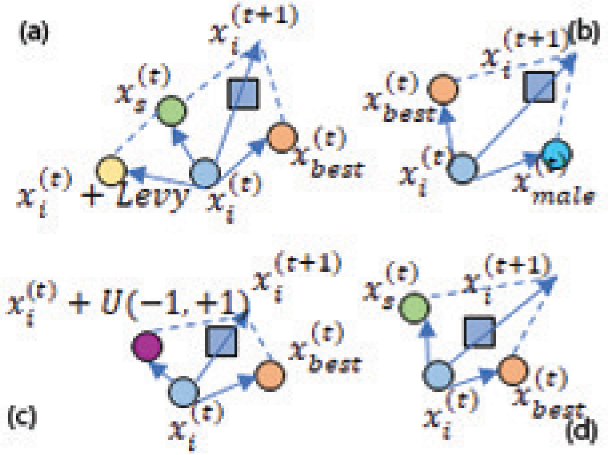

Fig. 1. Movement trajectories of BIS agents.

c) Learning process of agents: Fig. 1 illustrates the movement trajectories of male and female agents. Fig. 1(a) and Fig. 1(b) depicts movement of single and mated females and similarly, Fig. 1(c) and Fig. 1(d) represents the movement of single and mated male agents. In Fig. 1(a), (x_s) represents the suggested position obtained from the social group the agent belongs to, and it contributes as a stochastic guide to the single female agents' movement. The (x_{male}) shown in Fig. 1(b) represents the male mate of the female agent and the female agent is attracted towards the (x_{male}) while also moving towards the (x_{best}). The movement trajectory of the mated male shown in Fig. 1(d) is similar to the mated female's movement, but instead it uses its social group suggestion (or social learning) which results in the male agent exploiting the best-known position while performing a local search within the promising region based on the group suggestions. Lastly, in Fig. 1(c), a single male agent moves towards (x_{best}) and $\mathcal{U}(-r_5, +r_6)) \odot \mathcal{U}(0,1) r_7 \odot \mathcal{U}(0,1)$ which solely focuses on exploiting the best-known solution. Preferably, $\mathcal{U}(-r_5, +r_6)$ could be adjusted to control the scope of local learning, where $\mathcal{U}$ is a uniform distribution, $r_j, j = 1..8$ are random variables in the range [0,1], and $\odot$ is the Hamard product operator.

D. Implementation of the BIS algorithm

The swarm comprises n agents distributed in the search space, a vector represents each agent's position, where $x_{i,j}$ represents the j^{th} dimension of the i^{th} agent. Initially, agents are randomly distributed in the search space using uniform distribution and the fitness of each agent is determined by the value of the cost function at its position. Male agents with mates are responsible for providing a better position for their female mates to keep incubation temperature above a certain value. As mentioned previously, the female agent's temperature determines the gender of their offspring. $T_i^{(t)}$ is a scalar representing temperature for the i^{th} female agent at time t and is updated using:

$$T_i^{(t+1)} = \begin{cases} T_i^{(t)} + P & \text{if } f(x_i^{(t)}) < f(x_i^{(t-1)}) \\ T_i^{(t)} - P & \text{if } f(x_i^{(t)}) \geq f(x_i^{(t-1)}) \end{cases} \tag{1}$$

where f is the objective/fitness function, P is a penalty term used to determine the level of increase/decrease of temperature, calculated as

$$P = log\big(\big|f(x_i^{(t)}) - f(x_{mate})\big|\big)\lambda \tag{2}$$

where λ is a scaling factor used to control fluctuation for the penalty function. The anti-aphrodisiac is updated at the same time as temperature using

$$A_i^{(t+1)} = \begin{cases} U(0.5, 2) & \text{if } f(x_i^{(t)}) \leq \big(f(x_{best}) + f(x_{worst})\big)0.5 \\ U(0, 0.5) & \text{else} \end{cases} \tag{3}$$

where x_{best} and x_{worst} are position of agents with the best and worst fitness. At the time of reproduction, the gender of an offspring agent is determined using

$$G_i^{(t+1)} = \begin{cases} female & \text{if } T_i^{(t} < \alpha \\ male & \text{if } T_i^{(t)} \geq \alpha \end{cases} \tag{4}$$

where α is the threshold temperature to determine gender distinction. Agents are assigned hierarchies based on their mate status and the quality of solution found. As mentioned previously, hierarchies are used to create a subgroup of agents that are in similar situations. Each of these subgroups provides a suggested position for their own members represented as x_s and is calculated as:

$$x_s^{(t+1)} = \begin{cases} x_{best}^{(k)} \odot U(0, 1) & \text{if } \kappa > 2 \\ LB + U(0, 1) \times (UB - LB) & \text{else} \end{cases} \tag{5}$$

where x_i^k is the position of the i^{th} agent within the hierarchy k, x_{best}^k is the position of the best agent within the hierarchy k and κ is the size of the social group. The movement of agents depends on their mate status, gender and maturity status. Female agents with mates move towards their mate and the best-known position in an attempt to perform a local search using:

$$x_i^{(t+1)} = x_i^{(t)} + (x_{best} - x_i)\, r_1 + \big(x_i^{mate} - x_i\big) \odot \mathcal{U}\,(0, 1) \tag{6}$$

On the contrary, single female agents' movements are influenced by the suggestion of the social group combined with Lévy flights. Although single female agents have the tendency to move towards the x_{best}, the Lévy flights and the group suggestion enables more stochastic search to find a better position using:

$$x_i^{(t+1)} = x_i^{(t)} + (x_s + x_i + Levy(\theta) \odot \mathcal{U}\,(0, 1)) + (x_{best} - x_i)\, r_2 \tag{7}$$

The movement of male agents with mates are influenced by the social group suggestion to explore potentially promising local regions while moving towards the x_{best} using the update equation:

$$x_i^{(t+1)} = x_i^{(t)} + (x_{best} - x_i)\, r_3 + (x_s - x_i) \odot \mathcal{U}\,(0, 1) \tag{8}$$

Single male agents merely focus on exploiting the best-known solution; hence its movement is limited to taking local steps around the x_{best}. The uniform distribution $\mathcal{U}(-r_5, +r_6)$ determines agent's scope of search and could be set relative to the problem bounds and linearly decreased with time to narrow down the scope of the search. The position of a single male agent is calculated using:

$$x_i^{(t+1)} = x_i^{(t)} + (x_{best} - x_i) + (\mathcal{U}\,(-r_5, r_6)) \odot \mathcal{U}\,(0, 1)\, r_7 \odot \mathcal{U}\,(0, 1))r_8 \tag{9}$$

Offspring agents' positions are highly influenced by parent 1(mother) and parent 2 (father). Post-reproduction, the record of the father is stored as x_{p2} and is used in the offspring's position update. A mother is selected from the pool of available single females represented as x_{p1}. Segment of size k from both parents are chosen where $k = \lfloor d/2 \rfloor$ and added together to form a new vector v. A random integer q is generated to determine the number of dimensions (d) to modify in v and subsequently J_r (random index) is generated by permutation to randomly select which dimensions of v to modify.

$$v = (x_{p1}, x_{p2})\,, q \in [0, d]\,, J_r = randperm\,(d, q) \tag{10}$$

Finally, the offspring position is updated by element-wise multiplication of the j_r^{th} dimension of v and a random vector with uniform distribution using

$$x_i^{(t+1)} = for\ j = 1 : dim\,(J_r)\,, \ldots, v_{J_r} \odot \mathcal{U}\,(0, 1) \tag{11}$$

Agents pursuing a mating strategy to find a suitable mate are required to select from a set of available agents. In this case, mean absolute error is used to measure the rate of dissimilarity between the selector and the selected candidate. The fundamental reason for enforcing mating between the most dissimilar agents is to maintain diversity within the swarm as an offspring's position is elicited from both parents.

III. Results and Discussions

A. Experimental Setup

To verify the performance of BIS on different problem domains, 5 distinct experiments were conducted with various problem types and complexities. Experiment E1 (100-dimensional), E2 (500-dimensional) and E3 (1000-dimensional) were conducted using 40 unconstrained benchmark problems shown in Table I, and E4 using (100-dimensional) CEC2017 test suite, and E5 using 5 well-known real-world problems namely, (speed reducer design, tension-spring design, pressure vessel design, welded beam design and three-bar truss design problem). Besides BIS, comparisons

Algorithm 1: BIS Algorithm

```
see supplementary material for the parameter settings.
for t=1:$T_{max}$ do
    for i=1:pop do
        if agent is an offspring then
            update the position using Eqs.13 and 14;
        end
        if agent is female, mature, has a mate then
            if time to reproduce then
                calculate offspring's gender using Eq.4;
                update the position using Eqs.10-11;
            else
                update the position using Eq. 6;
            end
            if time to calculate penalty and temp. then
                calculate the temprature using Eq.1;
                calculate the penalty using Eq.2;
                update the anaphrodisiac using Eq.3;
                if current anaphrodisiac level $< \beta$ then
                    abandon the current mate;
                end
            end
        end
        if agent is female, mature and single then
            update the position using Eq.7;
        end
        if agent is male, mature and mated then
            update the position using Eq.9;
        end
        if agent is male, mature and single then
            update the position using Eq.8;
        end
    end
    if t < ($T_{max}$/2) then
        $\theta$ = $\theta_{max}$ - ($\theta_{min}$ - $\theta_{max}$) * t/$T_{max}$;
    end
end
```

were conducted on 6 well-established meta-heuristics including : bat algorithm (BA) with parameters [27] $A = 0.25, r = 0.5, f_{min}, f_{max} = 0, 2$, cuckoo search algorithm (CS) [33] p=0.25, differential evolution (DE) [34] $F = 0.7$, $C_r = 0.9$, firefly algorithm (FA) [28] $\alpha = 0.5, \beta = 0.2, \gamma = 1$, flower pollination (FP) [30] $p = 0.8$, and inertia weight particle swarm optimisation (PSO) [31] $\omega = 0.9 \rightarrow 0.4, c_1, c_2 = 2$, and *4 recent variants* including firefly algorithm with adaptive control parameters (AfPA) [34] $\alpha(0) = 0.5, \beta_0(0) = 1, \alpha(t)$ and $\beta_0(t) = adaptive, \gamma = 1$, cloud mutation flower pollination algorithm (CMFPA) [35] $p = 0.8$, opposition-based learning modified particle swarm optimisation (OLMPSO) [36] $\omega = dynamic, c_1, c_2 = 2.05$ and self-adaptive dual-strategy differential evolution (SaDSDE) [37] $F_i^t = (f_{max}^t - f(X_i^t))/(f_{max}^t - f_{min}^t), \lambda = 1 - cos((t/T)^2)$. Each algorithm used a fixed set of parameters which had previously been found to be the best general setting to give good results over a range of problems and each problem was run 100 times for 150,000 function evaluations with a population of 30. Each algorithm was reimplemented in the same framework as BIS, and results were checked against code available from the algorithms' original authors (in all cases the reimplemented versions gave exactly the same results as the original code). All problems used continuous variables. An external supplementary material is provided for detailed problem definitions and results of experiments that can be accessed from users.sussex.ac.uk/fv47/BIS.pdf.

TABLE I
THE BENCHMARK PROBLEMS USED FOR EXPERIMENTS E1, E2 AND E3.

#	Function	Range
F1	Ackley 1 Function	$x_i \in [-32.768, 32.768]$
F2	Griewank Function	$x_i \in [-600, 600]$
F3	Levy Function	$x_i \in [-10, 10]$
F4	Rastrigin Function	$x_i \in [-5.12, 5.21]$
F5	Schwefel Function	$x_i \in [-500, 500]$
F6	Sphere Function	$x_i \in [-5.12, 5.21]$
F7	Sum of Different Powers Function	$x_i \in [-1, 1]$
F8	Sum Squares Function	$x_i \in [-10, 10]$
F9	Trid Function	$x_i \in [-d, d]$
F10	Dixon-Price Function	$x_i \in [-10, 10]$
F11	Rosenbrock Function	$x_i \in [-5, 10]$
F12	Michalewicz Function	$x_i \in [0, \pi]$
F13	Powell Function	$x_i \in [-4, 5]$
F14	Styblinski-Tang Function	$x_i \in [-5, 5]$
F15	Ackley N.4	$x_i \in [-35, 35]$
F16	Alpine N.1	$x_i \in [0, 10]$
F17	Exponential	$x_i \in [-1, 1]$
F18	Powell Sum	$x_i \in [-1, 1]$
F19	Qing	$x_i \in [-500, 500]$
F20	Quartic	$x_i \in [-1.28, 1.28]$
F21	Salomon	$x_i \in [-100, 100]$
F22	Schwefel 2.20	$x_i \in [-100, 100]$
F23	Schwefel 2.23	$x_i \in [-10, 10]$
F24	Shubert 3	$x_i \in [-10, 10]$
F25	Shubert 4	$x_i \in [-10, 10]$
F26	Xin-She Yang N.2 Function	$x_i \in [-2\pi, 2\pi]$
F27	Xin-She Yang N.4 Function	$x_i \in [-10, 10]$
F28	Brown Function	$x_i \in [-1, 4]$
F29	Happy Cat Function	$x_i \in [-2, 2]$
F30	Ridge Function	$x_i \in [-5, 5]$
F31	Schwefel 2.21	$x_i \in [-100, 100]$
F32	Chung Reynolds Function	$x_i \in [-100, 100]$
F33	Csendes Function	$x_i \in [-1, 1]$
F34	Deb 1 Function	$x_i \in [-1, 1]$
F35	Deb 3 Function	$x_i \in [-1, 1]$
F36	Quintic Function	$x_i \in [-10, 10]$
F37	Schumer Steiglitz Function	$x_i \in [-100, 100]$
F38	Step Function	$x_i \in [-100, 100]$
F39	Step 2 Function	$x_i \in [-100, 100]$
F40	Stepint Function	$x_i \in [-5.12, 5.12]$

B. Results

For E1, the BIS algorithm was superior in the majority of the unconstrained test problems. As shown in Fig. 2(a), for 22 of the 40 problems, BIS found the best solution. SaDSDE showed comparable performance by finding best solutions for the same number of problems. As shown in Fig. 3(a), for the same problem set, the global optimum was discovered by BIS for 21 problems revealing that the

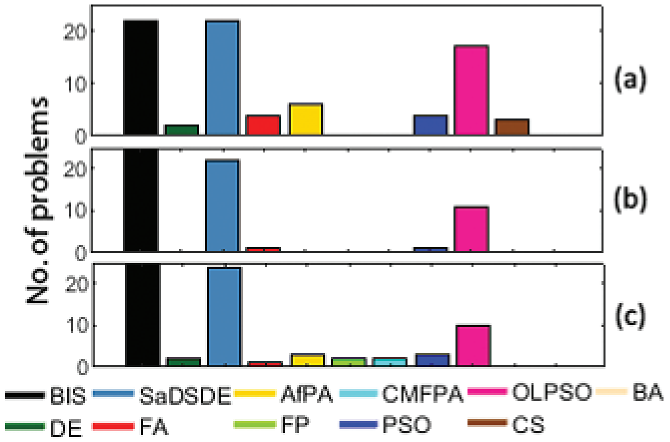

Fig. 2. Number of best solutions found for E1 (a), E2 (b), E3 (c).

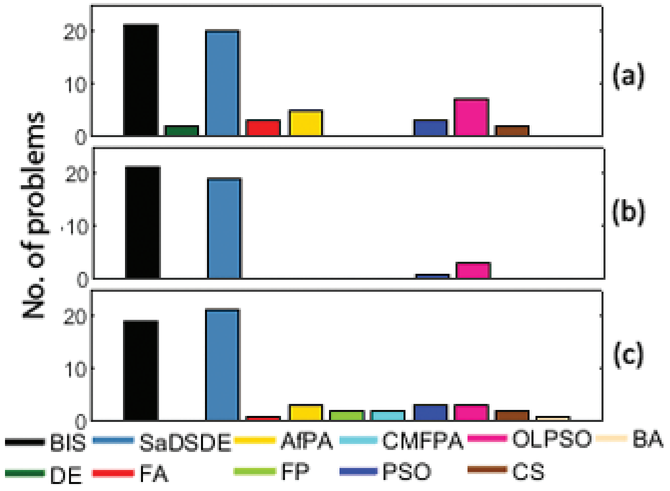

Fig. 3. Number of global optimums found for E1 (a), E2 (b), E3 (c)

proposed algorithm converged to the global optimum for the majority of the 100-dimensional problems. Considering the 22 problems where BIS obtained the best solution among other methods, only in 1 case did it find a near-optimal solution, converging to the global optimum for 21 of the 22 problems solved. Overall performance for E2 shown in Fig. 2(b) reveals that BIS exhibited the top performance by finding the best solutions for 25 problems. It was observed that DE, FA, AfPA, PSO and CS exhibited deterioration in performance with increased dimensionality, finding fewer or no best solutions in comparison to experiment E1. As shown in Fig. 3(b), BIS converged to global optima for 21 of the 25 problems it found

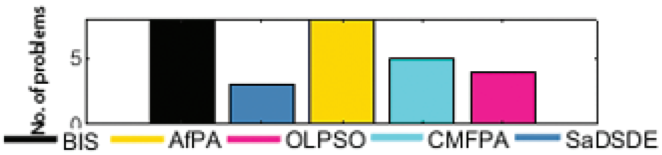

Fig. 4. Number of best solutions found for the CEC'17 test suite.

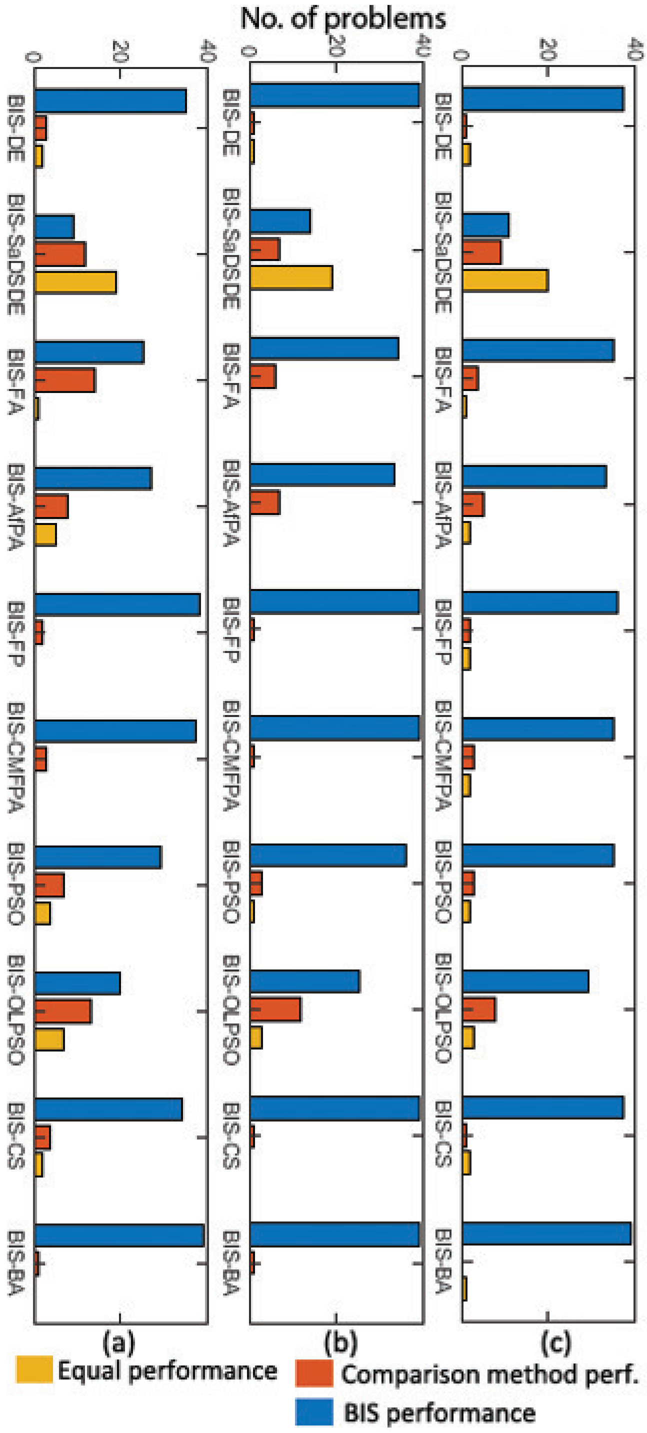

Fig. 5. Pairwise performance comparison for E1 (a) E2, (b) and E3 (c).

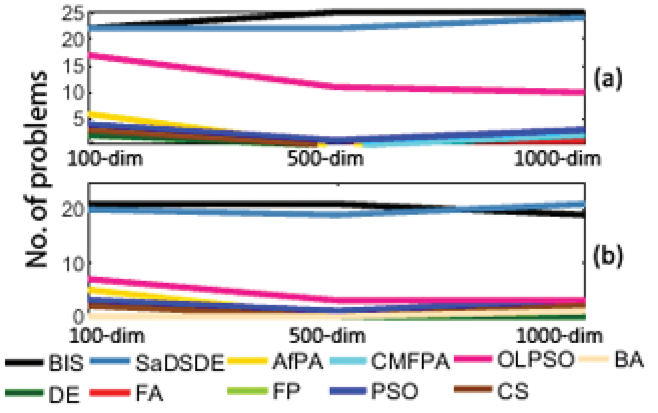

Fig. 6. Performance deteriorations with increased dimensionality in finding the best solutions (a) and global optimum discovery (b).

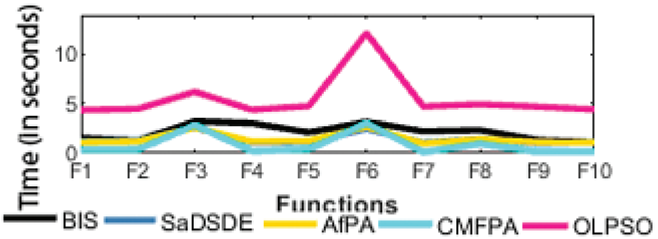

Fig. 7. Run time comparison of BIS and 4 recent meta-heuristics.

best solutions for, revealing the global optimum discover for majority of problems despite increased dimensionality. Fig. 2(c) shows the the best performances for E3. BIS found the best solutions among comparison methods for 25 problems. It can be seen from Fig. 3(c) that BIS converged to global optima for 19 of the 25 problems it solved and SaDSDE discovered the global optima for 21 problems. This is the only case where BIS is outperformed by another method in finding global optimum. It is also worth noting that, on nearly all other problems in this set, BIS found very good near optimal solutions. To further test the performance of the BIS, experiment E4 is conducted using (100-dimensional) CEC'17 benchmark suite, an up-to-date one of the most challenging test suite employed for numerical optimisation competitions. As shown in Fig. 4, BIS outperforms all compared methods in a total of 8 of the 28 problems. A comparable performance is observed from AfPA which found best near-optimal solutions for the same number of problems. The results for E5 reveal that BIS exhibited very competitive performance and discovered the global optimum for 3 of the 5 real-world problems and found very good near-optimal solutions for the other 2 problems. In experiments E1-E3, the most competitive performance against BIS was achieved by SaDSDE, however it's observed that on a more challenging CEC17 test suite, the performance of SaDSDE shows significant deterioration (unlike BIS) and exhibits the least optimal performance by outperforming comparison methods in only 3 problems. Fig. 5 shows the pairwise performance comparison of BIS with the other methods for unconstrained 100, 500 and 1000-dimensional problems. The comparison reveals that for 100-dimensional problems, as shown in Fig. 5(a), SaDSDE is the only algorithm to exhibit the best performance against the proposed method among other methods by outperforming BIS in 12 problems while BIS outperforms SaDSDE in 9 cases. For 500 and 1000 dimensional problems, shown in Fig. 5(b) and Fig. 5(c), visible deterioration is observed in the performance of comparison algorithms and BIS outperformed all comparison algorithms in both cases. It's worth noting that, as observed from Fig. 6(a) and Fig. 6(b), BIS is the sole method to avoid deterioration in finding best solution with increased dimensionality and complexity of problems. This pattern of growth in finding the most best solutions, along with minimal deterioration in global optimum discovery, results in excellent performance consistency. To measure the significance of the results obtained for E1, E2 and E3, the Friedman test was conducted on average performance of all algorithms, and the results reveal there is significant difference for E1 (p-value=4.1999e-10), E2 (p-value=7.8148e-11) and E3 (p-value=6.0600e-09). Post-hoc analysis [38] was carried out showing that the BIS's performance is significantly better in comparison to the other methods. The visible performance contrast of BIS compared to the other methods (especially DE, FA, FP, PSO, CS and BA) may seem controversial hence, to neutralize any dispute, an additional 2 experiments were carried out for these 6 algorithms with the 100-dimensional problems using other parametric settings [39]–[49] that are more 'tuned' to this problem subset. The results confirmed that BIS (with unchanged settings) remains superior. Runtime comparison were conducted between the 4 recent variants (AfPA, CMFPA, OLPSO and SaDSDE) and BIS; Fig. 7 shows the average run time (in seconds) obtained from 20 subsequent runs of each algorithm for ten 100-dimensional problems (F1-F10). BIS shows faster run time in comparison to SaDSDE in all cases and a comparable runtime to CMFPA, AfPA, OLPSO in some cases and in others with slight difference. However, considering the overall success of BIS, this minor runtime difference observed in some problems is easily tolerable.

IV. Conclusion

The BIS algorithm was shown to perform very well on a wide range of optimisation problems, including high dimensional and constrained problems. When compared with various canonical and recent extended variants of well-known metaheuristics, BIS outperforms these algorithms on most benchmark problem sets and exhibited highly competitive computational efficiency. The degeneracy afforded by multiple interacting mechanisms appears to assist and make BIS a robust optimisation algorithm. The present study may be further extended to enable the proposed algorithm to solve multi-objective and NP-hard combinatorial problems.

References

[1] Kennedy, J., Eberhart, R. and Shi, Y. (2001). Swarm intelligence. San Francisco: Morgan Kaufman Publishers.

[2] Yang, X., Gandomi, A. and Alavi, A. (2013). Metaheuristics in water, geotechnical and transport engineering. Amsterdam: Elsevier.

[3] Gandomi, A., Yang, X., Talatahari, S. and Alavi, A. (2013). Metaheuristic applications in structures and infrastructures. London: Elsevier.

[4] D. Câmara, A. Loureiro, GPS/Ant-like routing in ad hoc networks, Telecommun. Syst. 18 (1–3) (2001) 85–100.

[5] H. Sandalidis, K. Mavromoustakis, P. Stavroulakis, Performance measures of an ant based decentralized routing scheme for circuit switching communication networks, Soft Comput. 5 (4) (2001) 313–317.

[6] H. Sandalidis, K. Mavromoustakis, P. Stavroulakis, Ant-based probabilistic routing with pheromone and antipheromone mechanisms, Int. J. Commun. Syst. 17 (2004) 55–62.

[7] T. White, B. Pagurek, F. Oppacher, Connection management using adaptive mobile agents, in: H. Arabnia (Ed.), Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications, CSREA Press, Boston, MA, 1998, pp. 802–809.

[8] K. Vaisakh, P. Praveena, S.R.M. Rao, K. Meah, Solving dynamic economic dispatch problem with security constraints using bacterial foraging PSO-DE algorithm, Int. J. Electr. Power Energy Syst. 39 (1) (2012) 56–67.

[9] B. Panigrahi, V.R. Pandi, S. Das, Adaptive particle swarm optimization approach for static and dynamic economic load dispatch, Energy Conv. Manag. 49 (6) (2008) 1407 1415.

[10] Y. Wang, J. Zhou, Y. Lu, H. Qin, Y. Wang, Chaotic self-adaptive particle swarm optimization algorithm for dynamic economic dispatch problem with valve-point effects, Expert Syst. Appl. 38 (11) (2011) 14231–14237.

[11] X. Yuan, A. Su, Y. Yuan, H. Nie, L. Wang, An improved PSO for dynamic load dispatch of generators with valve-point effects, Energy 34 (1) (2009) 67–74.

[12] Wang, C. and Song, W. (2019). A novel firefly algorithm based on gender difference and its convergence. Applied Soft Computing, 80, pp.107-124.

[13] Nguyen, T., 2019. A high performance social spider optimization algorithm for optimal power flow solution with single objective optimization. Energy, 171, pp.218-240.

[14] Yazdani, M. and Jolai, F., 2015. Lion Optimization Algorithm (LOA): A nature-inspired metaheuristic algorithm. Journal of Computational Design and Engineering, 3(1), pp.24-36.

[15] Yang, X. (2013). Swarm intelligence and bio-inspired computation. 1st ed. London: Elsevier, p.49-50.

[16] Yang, X. (2013). Swarm intelligence and bio-inspired computation. 1st ed. London: Elsevier, p.70.

[17] Li, W., Meng, X., Huang, Y. and Fu, Z., 2020. Multipopulation cooperative particle swarm optimization with a mixed mutation strategy. Information Sciences,.

[18] Wood, D., 2016. Hybrid cuckoo search optimization algorithms applied to complex wellbore trajectories aided by dynamic, chaos-enhanced, fat-tailed distribution sampling and metaheuristic profiling. Journal of Natural Gas Science and Engineering, 34, pp.236-252.

[19] Husbands, P., Philippides, A. et al. (2010) Spatial, temporal and modulatory factors affecting GasNet evolvability in a visually guided robotics task, Complexity 16(2):35-44.

[20] Edelman, G. and Gally, J., 2001. Degeneracy and complexity in biological systems. Proceedings of the National Academy of Sciences, 98(24), pp.13763-13768.

[21] Whitacre, J. and Bender, A., 2010. Degeneracy: A design principle for achieving robustness and evolvability. Journal of Theoretical Biology, 263(1), pp.143-153.

[22] Warner, D. (2011). Sex Determination in Reptiles. Hormones and Reproduction of Vertebrates, pp.1-38.

[23] Lovern, M. (2011). Hormones and Reproductive Cycles in Lizards. Hormones and Reproduction of Vertebrates, pp.321-353.

[24] Drickamer, L. (2019). Mating Systems in Mammals. Encyclopedia of Animal Behavior, pp.533 538.

[25] Sivinski, J. (1984). Sperm in Competition. Sperm Competition and the Evolution of Animal Mating Systems, pp.85-115.

[26] Knowlton, N. and Greenwell, S. (1984). Male Sperm Competition Avoidance Mechanisms: The Influence of Female Interests. Sperm Competition and the Evolution of Animal Mating Systems, pp.61-84.

[27] Yang, X. (2016). Nature-inspired optimization algorithms.: Elsevier, pp.259.

[28] Yang, X. (2016). Nature-inspired optimization algorithms.: Elsevier, pp.250-251.

[29] Molugaram, K. and Rao, G. (2017). Chi-Square Distribution. Statistical Techniques for Transportation Engineering, pp.383-413.

[30] Yang, X. (2016). Nature-inspired optimization algorithms.: Elsevier, pp.261-263.

[31] Aliman, M., Ibrahim, Z., Naim, F., Nawawi, S. and Sudin, S. (2015). Performance evaluation of black hole algorithm, gravitational search algorithm and particle swarm optimization. Malaysian Journal of Fundamental and Applied Sciences, 11(1).

[32] Yang, X. (2014). Nature-Inspired Optimization Algorithms. Elsevier, pp.183-196.

[33] Yang, X. (2016). Nature-inspired optimization algorithms.: Elsevier, pp.256-258.

[34] Wang, H., Zhou, X., Sun, H., Yu, X., Zhao, J., Zhang, H. and Cui, L., 2017. Firefly algorithm with adaptive control parameters. Soft Computing, 21(17), pp.5091-5102.

[35] Chen, Y. and Pi, D., 2020. An innovative flower pollination algorithm for continuous optimization problem. Applied Mathematical Modelling, 83, pp.237-265.

[36] O. P. Verma, S. Gupta, S. Goswami and S. Jain, "Opposition based modified particle swarm optimization algorithm," 2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Delhi, 2017, pp. 1-6.

[37] Duan, M., Yang, H., Wang, S. and Liu, Y., 2019. Self-adaptive dual-strategy differential evolution algorithm. Plos One, 14(10), p.e0222706.

[38] D. Janez, S. Dale, Statistical comparisons of classifiers over multiple datasets, J. Mach. Learn. Res. 7 (2006) 1–30.

[39] Yang, X. (2016). Nature-inspired optimization algorithms.: Elsevier, pp.259.

[40] S. Bencharef, H. Boubertakh, "Optimal Tuning of a PD control by Bat Algorithm to Stabilize a Quadrotor", 8th Int. Conf. on Modelling Identification and Control, 2016.

[41] X.-S. Yang, S. Deb, Cuckoo search via Lévy flights, World Congress on Nature & Biologically Inspired Computing (NaBIC 2009). IEEE Publications, pp. 210-214, 2009.

[42] Pakzad-Moghaddam, S., Mina, H. and Mostafazadeh, P. (2019). A novel optimization booster algorithm. Computers & Industrial Engineering, 136, pp.591-613.

[43] Civicioglu, P., Besdok, E. A conceptual comparison of the Cuckoo-search, particle swarm optimization, differential evolution and artificial bee colony algorithms. Artif Intell Rev 39, 315–346 (2013).

[44] Wang, C. and Song, W. (2019). A novel firefly algorithm based on gender difference and its convergence. Applied Soft Computing, 80, pp.107-124.

[45] Kohli, M. and Arora, S. (2018). Chaotic grey wolf optimization algorithm for constrained optimization problems. Journal of Computational Design and Engineering, 5(4), pp.458-472.

[46] Yang XS. (2012) Flower Pollination Algorithm for Global Optimization. In: Durand-Lose J., Jonoska N. (eds) Unconventional Computation and Natural Computation. UCNC 2012. Lecture Notes in Computer Science, vol 7445. Springer, Berlin, Heidelberg.

[47] Benkercha, R., Moulahoum, S., Colak, I. and Taghezouit, B., 2016. PV module parameters extraction with maximum power point estimation based on flower pollination algorithm. 2016 IEEE International Power Electronics and Motion Control Conference (PEMC),.

[48] Jain, M., Singh, V. and Rani, A. (2019). A novel nature-inspired algorithm for optimization: Squirrel search algorithm. Swarm and Evolutionary Computation, 44, pp.148-175.

[49] Kohli, M. and Arora, S. (2018). Chaotic grey wolf optimization algorithm for constrained optimization problems. Journal of Computational Design and Engineering, 5(4), pp.458-472.

[50] R. C. Eberhart, and J. Kennedy, "A new optimizer using particle swarm theory", Proc. of the Sixth Int. Symposium on Micromachine and Human Science, Nagoya, Japan. pp. 39-43, 1995

[51] Storn, R. and Price, K., 1997. Journal of Global Optimization, 11(4), pp.341-359.

[52] M.R. Tanweer, S. Suresh, N. Sundararajan, Dynamic mentoring and self-regulation based particle swarm optimization algorithm for solving complex real-world optimization problems, Inform. Sci. 326 (2016) 1–24.