Neural Computing and Applications (2019) 31:8837–8857
https://doi.org/10.1007/s00521-019-04464-7

ORIGINAL ARTICLE

The naked mole-rat algorithm

Rohit Salgotra[1] · Urvinder Singh[1]

Received: 1 February 2017 / Accepted: 26 August 2019 / Published online: 11 September 2019

Abstract

This work proposes a new swarm intelligent nature-inspired algorithm called naked mole-rat (NMR) algorithm. This NMR algorithm mimics the mating patterns of NMRs present in nature. Two types of NMRs called workers and breeders are found to depict these patterns. Workers work continuously in the endeavor to become breeders, while breeders compete among themselves to mate with the queen. Those breeders who become sterile are pushed back to the worker's group, and the fittest worker becomes a new breeder. This phenomenon has been adapted to develop the NMR algorithm. The algorithm has been benchmarked on 27 well-known test functions, and its performance is evaluated by a comparative study with particle swarm optimization (PSO), grey wolf optimization (GWO), whale optimization algorithm (WOA), differential evolution (DE), gravitational search algorithm (GSA), fast evolutionary programming (FEP), bat algorithm (BA), flower pollination algorithm (FPA), and firefly algorithm (FA). The experimental results and statistical analysis prove that NMR algorithm is very competitive as compared to other state-of-the-art algorithms. The matlab code for NMR algorithm is avaliable at https://github.com/rohitsalgotra/Naked-Mole-Rat-Algorithm.

Keywords Swarm intelligence · Eusociality · Optimization · Benchmark · Naked mole-rat algorithm

1 Introduction

Nature has always served as a problem solver, and it has been solving real complex problems for millions of years. A large number of natural systems are working in collaboration to effectively achieve numerous tasks at hand. The mammalian body itself consists of a large number of systems working in a systematic way for performing various body functions, eyes as a perfect visual system, ears as a sound receptor, brain as the mastermind of all systems, and others. Animal species such as wolves show predator–prey system, honey bees showing cooperative behavior, fireflies attracting other fireflies for mating, moths moving toward the source of light, and whales searching for preys in water. In short, all natural species show some sort of intelligence, and artificial systems have been derived using their knowledge from the past two decades. The knowledge base has itself been developed into a separate field of study, and a large number of algorithms known as nature-inspired algorithms have been derived.

Nature-inspired algorithms are becoming extremely popular for solving various optimization problems. This is due to the fact that these algorithms search for the fittest solution based upon'trial and error' criterion. Also, they are easy to implement due to their simple conceptual model and the least requirement of gradient information. Broadly, these algorithms have been inspired by theories like Darwin's theory of natural selection and social behavior of living organisms. Darwin's theory led to the development of evolutionary algorithms (EAs), while social behavior paved the way for swarm intelligent (SI) algorithms. All EAs are stochastic, and hence require a minimum pool of population to improve their performance, while for SIs, new solutions are generated by incorporating appropriate operators, like the crossover, mutation, probability, and others.

Generally, EAs randomly select a population and evolve it over subsequent iterations. The best individuals of each iteration are combined together to form a new solution and hence optimizing all the solutions over subsequent iterations. These algorithms are best fit for solving problems where no initial experience is required and even without

✉ Rohit Salgotra
rohit.salgotra@thapar.edu

[1] Department of ECE, Thapar University, Patiala, Punjab, India

any specific knowledge about the problem under test. This is the reason that these algorithms have been applied to almost every domain of optimization. The examples of these algorithms are genetic algorithm (GA) [1], differential evolution (DE) [2], evolutionary strategy (ES) [3], biogeography-based optimization (BBO) [4], genetic programming (GP) [5], ant lion algorithm (ALO) [6], and probability-based incremental learning (PBIL) [7].

The second group of nature-inspired algorithms is SI. Swarm in itself means the group of different organisms, such as ants, birds, bees, monkeys, and wolves, which work collectively toward a common goal. SIs use the learning ability and adaptive nature of these organisms to solve real complex problems at hand. The first work on SI was reported by Beni and Wang [8] in cellular robotics. The field has flourished from that day, and today, these algorithms have been applied to almost every field whether it is electrical, management, electromagnetics, structural engineering design, and others. The most recent algorithms which have been applied to a lot of problems include artificial bee colony (ABC) [9], particle swarm optimization (PSO) [10], firefly algorithm (FA) [11], bacteria foraging optimization (BFO) [12], flower pollination algorithm (FPA) [13, 14], bat algorithm (BA) [15], and so on. Other recently introduced algorithms include salp swarm algorithm [16], dragonfly algorithm [17], cuckoo search [18, 19], hybrid GA [20], water wave optimization [21], whale optimization algorithm [22], moth flame optimization [23], and others.

The concept of SI was put forth by Bonabeau, and he defined it as '*an attempt to design algorithms or distributed problem-solving devices inspired by the collective behavior of social insects colonies and other animal societies*' [24]. To depict whether a behavior is SI or not, Karaboga et al. coined that for any distributed problem-solving system to adapt and self-organize, they should follow two fundamental principles, that is, self-organization and division of labor [9].

Self-organization is the process by which, low-level components interact with each other to produce a global result. The main feature is that there is no central control over the population, making all the elements to work and act at the same time. Hence, a global leader or solution is produced purely on the basis of local interactions. Self-organization relies on four basic principles [24]:

(a) Positive feedback: This phenomenon extracts information from the output of one system and reapplies it to input for producing convenient structures. It helps to provide diversity and accelerates the system. Examples include recruitment and reinforcement such as ant trails and bee dances.
(b) Negative feedback: This phase is to counterbalance the effects of positive feedback. So, it ultimately helps to stabilize the collective patterns and avoid saturation caused due to foragers, crowding, competition, or food exhaustion.
(c) Fluctuations: It refers to the randomness in a system, for innovation, creativity, and discovery of new solutions. This helps solution to get out of any stagnation problem.
(d) Multiple interactions: It provides different ways to learn based on interactions within a population. It thus enhances the intelligence by using combined results of all.

Division of labor: It means performing specialized tasks by individual groups, and this is the second phase to depict SI behavior for any organism. All the tasks are simultaneously performed, and hence the performance of such a system is better than the sequential tasks performed by unspecialized individuals [25, 26]. Division of labor also enables swarm to produce a response as per the changes in the search space.

Nature-inspired algorithms do have certain drawbacks. They may be fit for one problem, while unfit for others, or simply we can say that no algorithm is perfect for solving all optimization problems. This has been validated by the no free lunch theorem (NFL) [NMRs occur in semiarid]. This theory motivates many researchers to investigate and derive new algorithms for solving optimization problems. Another reason to design a new algorithm is the inherent drawbacks of existing algorithms like inefficiency in exploring search space by ABC [28], premature convergence, and stagnation of population-based algorithms like DE [29] and PSO [30]. PSO is also found to be weaker in refining the solution due to less diversity in later stages, and also problem-based tuning of parameters is required to get an optimal solution [31]. Also, there is a major defect in one of the most recently introduced grey wolf optimization algorithm [32]. This defect is the decrease in efficiency if the final solution tends to move away from zero as the final optimal solution. So, in order to eradicate these drawbacks, we need to evaluate, hybridize, and design new powerful algorithms for solving optimization problems. The main motive is to propose a generalized algorithm which can be extended to any field of research in optimization.

This paper proposes a new algorithm based on the mating patterns of naked mole-rat (NMR). The mating patterns of NMR show that these belong to the category of SI algorithms. So, the algorithm is proposed based on the mating pattern, eusociality, and SI behavior discussed in the subsequent section. Further, the proposed strategy is tested on various standard benchmark optimization test problems. The algorithm has also been tested statistically in order to prove its significance.

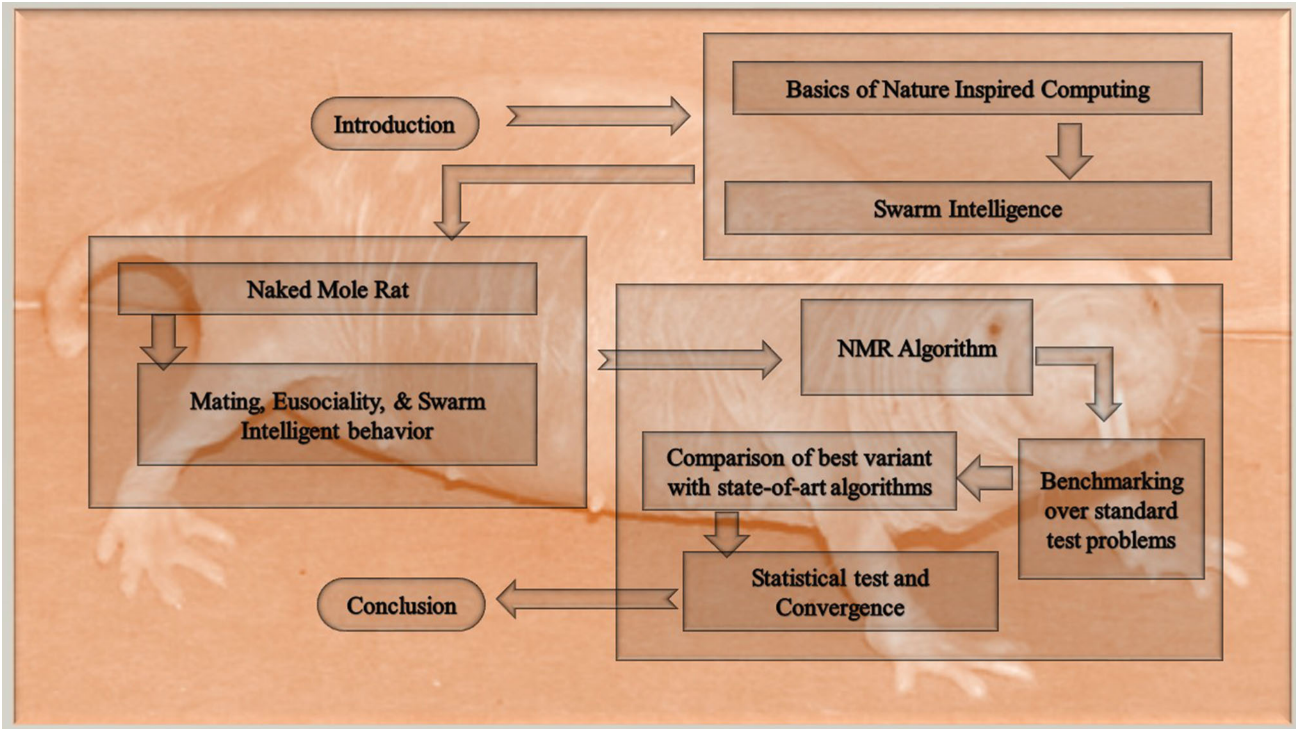


Fig. 1 Proposed pipeline of the conducted research

The rest of the paper is outlined as follows: Sect. 2 details about the basics of NMR algorithm; in Sect. 3, NMR is proposed; Sect. 4 has results and discussion part, showing the performance evaluation of proposed approach, and in Sect. 5, the conclusion is presented. The proposed pipeline of the conducted research is given in Fig. 1.

2 Basics of naked mole-rat

2.1 Naked mole-rat

Sexually reproducing animals are evolving mainly due to inbreeding [32]; this is due to the fact that sexual recombination helps to increase the variability of offspring's [33]. This helps to increase efficiency, remove deleterious alleles from different genes, and provide resistance from parasites [34]. This also helps species in adapting to the local conditions [35]. There is a type of inbreeding called close inbreeding which occurs when alternative males are not available. This type of phenomenon is very rare in natural species [36]. Fallow deer is one such animal showing a father–daughter mating [37] and has been confirmed by the genetic model [38]. But from the behavioral and paternity analysis, there is no conformity of such behavior [37]. The only other species showing inbreeding patterns is the NMR.

NMRs occur in semiarid eastern Africa and are fossorial, showing large subterranean tunnel systems while foraging for tubers [39] as shown in Fig. 2b. They have evolved and adapted to this hostile habitat [40] and are found to be present on this earth from the Miocene era (24 million years ago) [41]. NMRs were firstly described in African mammals by Eduard Ruppell in 1842. He named them 'different headed' (*Heterocephalus*) and smooth-skinned/hairless (*grabber*) because of their odd-shaped skull and the absence of flurry pelage as shown in Fig. 2a. They belong to the *kingdom:* Animalia, *phylum:* chordate, *order:* Rodentia, *family:* Bathyergidae and was the only species classified in *genus* Heterocephalus until the 1980s [42]. But till date, 15 species have been included in this genus [43]. NMRs, as discussed, are close inbreeding animals and they do so in a belief that new colonies can be formed through fission, helping in sealing the tunnels [44]. By sealing the tunnels, the danger due to predation, extreme temperatures, and starvation are minimized. The subsequent subsections provide a brief overview of NMR behavior.

2.2 Mating patterns in NMR

They are eusocial animals living in a group of up to 295 individuals. A detailed discussion about eusociality in NMR is given in the next subsection. The average population of a single colony is through close to 75 [44]. The colony is headed by a single female, that is, there is only a single breeding female called queen. This queen breeds with a limited number of males, while the remaining

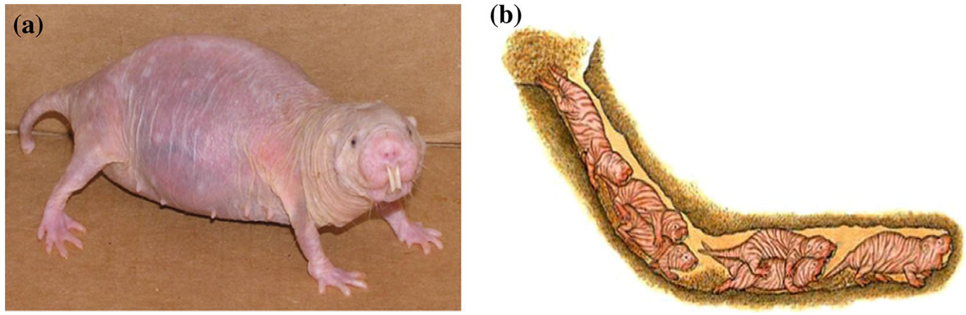


Fig. 2 **a** Naked mole-rat (NMR). **b** Burrowing pattern of NMR

members of the colony perform necessary tasks like construction, provisioning, maintenance, and defense. Workers are sterile and become sexually active only when separated from the colony. They separate from the colony only to act in the assistance of offspring produced by the queen [45]. So, based upon the mating patterns, we can define two types of NMRs that are breeders (assist in mating) and workers (helping to perform other tasks). Breeders live close to the queen and hence are at no risk or at lower risks of predation, while workers are not. This is the reason why breeders live up to 17 years, 4 times longer than their nonbreeding counterparts or workers [46]. So, is there only a single female in the colony that grows to a queen, if so, how will the new queen be selected if the present one die? The answer is that there are a large number of females in the colony, and they form the worker class. Any female who is older than 6 months is capable of becoming a queen. This further adds a question mark. Where will the present queen go, if any female can be queen? All the females in the group may fight until death to become the queen and establish their dominance [47]. When a female becomes the queen, it becomes reproductively active and starts producing estrogen-dependent 'pubescent' growth surge resulting in an increase in their body lengths [48]. The other females in the colony remain anovulatory with a lower level of sex steroids [49]. Males in the worker groups exhibit the same patterns and have low testosterone, low luteinizing hormone, abnormal sperms, and infertile [50]. Hence, it can be said that only the best males among the worker class form the breeder class and one among the best females get the status of a queen. The breeding mates, breeders, and workers are shown in Fig. 3.

2.2.1 Eusociality in NMR

Eusociality is defined as the colonial lifestyle of a species, following a strict division of labor and having a single female for breeding. This concept was laid down by Richard Alexander in the mid 1970s. He identified that such systems are found in the subterranean population living in soil and relying on others to dig burrows and forage food for underground tubers [51]. This theory was explored by Jarvis, who found that termites, ants, and wasps carry out tasks in a single community of about 295 individuals, with 1 to 3 breeding males and only a single breeding female [52]. So, it is evident from the study that NMRs are eusocial animals following the division of labor to perform various tasks and having only a single female called queen for mating. Division of labor is followed only by the workers, whereas breeders and queen are only meant for reproduction. This eusocial lifestyle of NMRs improves their inclusive fitness and ultimately contributes to the prolonged lifespan.

2.2.2 NMR and SI

Since the main focus of the paper is to design a new SI algorithm, so it is necessary to find out if that particular species is SI or not. In this subsection, we will try to find out if there is any close association of SI with NMR. So, if NMR will follow the necessary conditions of self-organization and division of labor, we can count it as SI animal. As explained:

A. Self-organization: It means a species should produce a global level response from low-level interactions without any central authority. In NMR, the workers act as a low-level component, and they produce high-level breeder. These high-level breeders further choose among themselves the best breeder, who mates with the queen. So, a hierarchical pattern is followed without any central authority. Hence, we can say that NMRs follow self-organization. This can be further validated by proving the four steps of self-organization as:

   (i) Positive feedback: In NMR, the queen is assisted by a few sets of breeding males and only they contribute to the reproduction process. These breeding males are selected from the pool of worker population. This process from workers to breeders and ultimately to queen follows positive feedback. This helps in providing diversity and accelerating the system to move to a new stable state.

   (ii) Negative feedback: This part focuses on compensation through positive feedback. The

Fig. 3 **a** Breeding mates. **b** Breeders. **c**. Workers

breeder, who is not able to cooperate or whose fitness is not up to the mark for mating, either dies or is sent back to the worker's pool, hence forming a negative feedback path from breeders to workers. This process stabilizes the system and hence maintains a proper level of breeder population.

(iii) Fluctuations: It means randomness in swarms. In NMR, randomness is in finding the best possible breeders and queen among the workers. This process helps in getting rid of the stagnation problem.

(iv) Multiple Interactions: It is a learning process either by interactions or experience. Here, the worker pool of NMRs interact with each other and cooperate or assist each other in completing various tasks like maintenance, construction, defense, and provisioning. The breeders also follow this pattern by cooperating with each other during mating, as only a single male with a mate with the queen at a particular time and hence enhance the combined intelligence of all NMRs.

B. Division of labor: It has been already discussed that NMRs are eusocial animals and every eusocial animal follows the strict division of labor. Here, the workers basically do all sorts of work and breeders do the mating.

From the above discussion, it is evident that NMRs are SI animals. In the next section, a new SI-based algorithm called NMRA has been proposed.

3 Naked mole-rat algorithm

The social behavior of NMR has inspired authors to propose a stochastic optimization algorithm. The algorithm mimics the mating patterns in NMR and has the following key features:

(i) NMRs are eusocial animals living in a group of 295 members with an average number of members to be 70–80. For experimental analysis, the present work uses a group of 50 NMRs.

(ii) A female queen leads the group and divides the population among breeders and workers. The breeders are the best performing NMR among the working group and are meant for mating only, whereas workers perform other tasks.

(iii) The workers perform necessary tasks and best among them are replaced by breeders. In simple words, high-performing workers become breeders, and low-performing breeders are again sent to the worker's pool.

(iv) The best breeder among the breeder pool mates with the queen.

The above four rules have been idealized to propose a naked mole-rat algorithm (NMR). The algorithm is divided into three phases. In the first phase, the population of NMRs is initialized, second is the worker phase, and third is the breeder phase. The breeder phase is selected based upon the breeding probability. The details of the algorithm are explained as:

(a) Initialization: Initially, it generates a uniformly distributed random population of n NMR where each NMR in the range [1, 2 … n] is a D-dimensional vector. Here, D represents the number of variables or parameters to be tested in the problem. Each NMR is initialized as

$$NMR_{i,j} = NMR_{min,j} + U(0,1) \times \left(NMR_{min,j} - NMR_{max,j}\right) \quad (1)$$

where $i \in [1, 2, \ldots n]$, $j \in [1, 2, \ldots D]$, $NMR_{i,j}$ is the ith solution in the jth dimension, $NMR_{min,j}$, $NMR_{max,j}$ are the lower and upper bounds of the problem function, respectively, and $U(0,1)$ is uniformly distributed random number. After initialization, the objective function is evaluated and its fitness is calculated. Based upon the fitness, B breeders and W workers are identified and overall initial best solution d is calculated. After initialization, the population of NMR is subjected to repeated cycles or iterations of the search process of worker and breeder phase.

(b) Worker phase: In this phase, the workers tend to improve their fitness so that they get a chance to become a breeder and eventually mate with the queen. So here, the new solution of worker NMR is generated based upon its own experience and local information. Here, the fitness of new NMR is evaluated, and if the new mating fitness is better, the old solution is discarded, and the new solution is memorized. Otherwise, the older solution is retained. After all the worker rats complete the search process, the final fitness of all of them is remembered. In order to produce a new solution from the old one, the NMR uses the following equation:

$$w_i^{t+1} = w_i^t + \lambda\left(w_j^t - w_k^t\right) \tag{2}$$

where w_i^t corresponds to the ith worker in the tth iteration, w_i^{t+1} is the new solution or worker, λ is the mating factor and w_j^t and w_k^t are two random solutions chosen from the worker's pool. The value of λ is obtained from a uniform distribution in the range of [0, 1].

(c) Breeder phase: The breeder NMR also update themselves in order to be selected for mating and also to stay as a breeder. The breeder NMRs are updated based upon a breeding probability (bp) with respect to the overall best d. This bp is a random number in the range of [0, 1]. Some of the breeders may not be able to update their fitness and hence may be pushed back to the workers' category. The breeders modify their positions according to the equation:

$$b_i^{t+1} = (1-\lambda)b_i^t + \lambda\left(d - b_i^t\right). \tag{3}$$

Here, b_i^t corresponds to the breeder i in the iteration t, λ factor controls the mating frequency of breeders and helps in identifying a new breeder b_i^{t+1} in the next iteration. To start with, the value of bp has been set to 0.5 as the initial value.

For simplicity, we have assumed that there is only a single queen, and the best among the breeder mates with the queen. So here, we find only the best breeding male who will breed with the female. The algorithm works by differentiating or identifying the breeders and workers among the pool of NMRs. After an initial evaluation, the best breeder and the best worker are selected. The fitness of workers is updated so that their fitness improves and they may get a chance to become breeders. On the other hand, breeders also update their fitness based upon breeding probability so that they remain breeders. The breeder which becomes sterile will be pushed into the workers' category. The best breeder among the population serves as the potential solution to the problem under test. The above-mentioned worker–breeder relationship and their mating with the queen are best elaborated in Fig. 4. The pseudo-code for NMRA is given in Algorithm 1.

```
Begin:
        Inputs: Initialize naked mole rats: n
                breeders B: n/5
                workers W: B − n
                define breeding probability: bp
                define D-dimensional objective function, f(x)
        Output: find the overall best d
    do Until iteration < maximum number of iterations
                for i=1: W
                        perform worker phase: w_i^{t+1} = w_i^t + λ (w_k^t − w_j^t)
                        evaluate w_i^{t+1}
                end for
                for i=1:B
                        if U(0,1) > bp
                        perform breeder phase: b_i^{t+1} = (1 − λ)b_i^t + λ(d − b_i^t)
                        evaluate b_i^{t+1}
                end for
                        combine the new worker and breeder population
                        evaluate the population
                        update the overall best d
                update iteration count
     end until
        save the final best (d)
End
```

Algorithm 1: Pseudo-code of Naked Mole Rat Algorithm

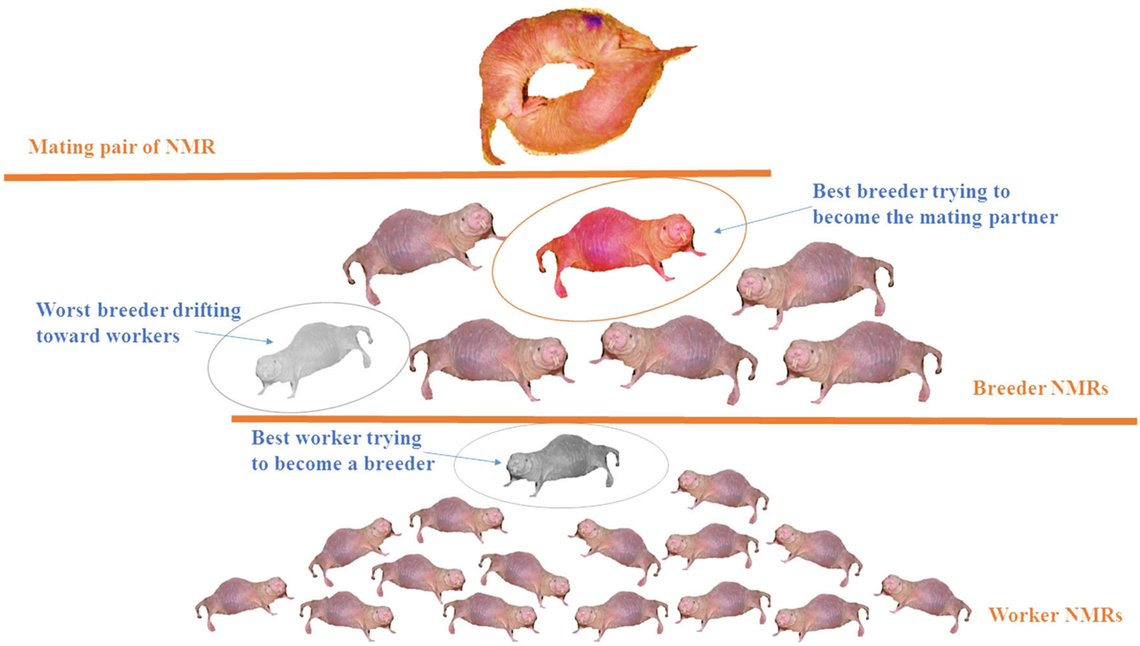


Fig. 4 Worker–breeder relationship and their mating pair

4 Result and discussion

A twofold strategy is followed for experimental analysis. Firstly, we aim to show that the proposed approach gives better results for the test suite when compared to standard algorithms, and then perform the rank-sum test to see if the proposed results are significant or not. The experimental study is divided into the following groups:

- Benchmarking
- Statistical testing.

For evaluating the fitness of the proposed approach, a total of 50 runs have been performed, and the following measures have been used:

- Best solution over 50 runs
- Worst solution after 50 runs
- Mean of 50 runs
- The standard deviation of 50 runs
- Statistical analysis by Wilcoxon rank-sum test.
- For comparison with results from the literature, 30 runs have been done, the same as reported by other authors.

In the rest of the paper, basic configuration of PC, test suite description, and results in terms of test suite have been presented.

4.1 PC configuration

All runs were made on Dell Inspiron 1464, with the following configuration:

- Processor: Intel (R) Core (TM) i3 CPU, M330 @ 2.13 GHz 2.13 GHz.
- RAM: 4.00 GB.
- Operating System: Windows 10, x64-based processor.
- MATLAB version: 7.10.0 R2010a.

4.2 Test suite

Table 1 shows full sets of test functions along with their dimensions, search range, and optimal solution. In order to evaluate the performance of NMRA, it has been applied to a standard set of benchmark function that is divided into three classes:

(i) *Unimodal functions:* These functions have no local solution and have only a single global solution. These benchmark functions are useful to examine heuristic optimization algorithms in terms of convergence rate.

(ii) *Multimodal functions:* These benchmark functions have many local minima that are increasing in number exponentially with dimension, so they are suitable to benchmark the capability of algorithms in avoiding local minima.

(iii) *Fixed dimension functions:* These functions show the consistency of an algorithm in finding optimum solutions.

The functions are shown in Table 1; here, function with sign *U* are unimodal, *M* are multimodal, and *FD* are fixed dimension functions.

Table 1 Description of test problems

Test problems	Objective function	Search range	Optimum value	D
Ackley function (M)	$f_1(x) = -20\exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2}\right) - \exp\left(\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi x_i)\right) + 20 + e$	[− 100, 100]	0	30
Colville function (FD)	$f_2(x) = 100\left(x_1^2 - x_2\right)^2 + (x_1 - 1)^2 + (x_3 - 1)^2 + 90\left(x_3^2 - x_4\right)^2 + 10.1\left((x_2 - 1)^2 + (x_4 - 1)^2\right) + 19.8(x_2 - 1)(x_4 - 1)$	[− 10, 10]	0	4
De Jong function N. 5 (M)	$f_3(x) = \left(0.002 + \sum_{i=1}^{25} 1 \Big/ i + (x_1 - a_{1i})^6 + (x_2 - a_{2i})^6\right)^{-1}$	[− 65.536, 65.536]	0	2
Dixon and Price (U)	$f_4(x) = (x_1 - 1)^2 + \sum_{i=2}^{d} i(2x_i^2 - x_{i-1})^2$	[− 10, 10]	0	30
Michalewicz (M)	$f_5(x) = -\sum_{i=1}^{D}\sin(x_i)\sin^{2m}\left(\frac{ix_i^2}{\pi}\right)$	[0, π]	−9.6601	10
Powell function (M)	$f_6(x) = \sum_{i=1}^{d}\left[(x_{4i-3} - 10x_{4i-2})^2 + 5(x_{4i-1} - x_{4i})^2 + (x_{4i-2} - 2x_{4i-1})^4 + 10(x_{4i-3} - x_{4i})^2\right]$	[− 4, 5]	0	30
Quartic (U)	$f_7(x) = \sum_{i=1}^{d} ix_i^4 + random[0, 1]$	[− 1.28, 1.28]	0	30
Salomon (M)	$f_8(x) = 1 - \cos\left(2\pi\sqrt{\sum_{i=1}^{D} x_i^2}\right) + \left(0.1\sqrt{\sum_{i=1}^{D} x_i^2}\right)$	[− 100, 100]	0	30
Sphere function (U)	$f_9(x) = \sum_{i=1}^{D} x_i^2$	[− 100, 100]	0	30
Step function (U)	$f_{10}(x) = \sum_{i=1}^{D}(\lvert x_i + 0.5\rvert)^2$	[− 100, 100]	0	30
Sum of different powers function (U)	$f_{11}(x) = \sum_{i=1}^{D}\lvert x_i\rvert^{i+1}$	[− 1, 1]	0	30
Tablet function (U)	$f_{12}(x) = 10^6 x_i^2 + \sum_{i=1}^{D} x_i^2$	[− 5, 5]	0	30
Zakharov function (M)	$f_{13}(x) = \sum_{i=1}^{d} x_i^2 + \left(\sum_{i=1}^{d} 0.5ix_i\right)^2 + \left(\sum_{i=1}^{d} 0.5ix_i\right)^4$	[− 5, 10]	0	30
Beale function (FD)	$f_{14}(x) = [1.5 - x_1(1 - x_2)]^2 + \left[2.25 - x_1\left(1 - x_2^2\right)\right]^2 + \left[2.625 - x_1\left(1 - x_2^3\right)\right]^2$	[− 4.5, 4.5]	0	2
Cigar function (M)	$f_{15}(x) = x_0^2 + 10000\sum_{i=1}^{D} x_i^2$	[− 10, 10]	0	30
Easom function (U)	$f_{16}(x) = -\cos x_1 \cos x_2 e^{\left(-(x_1-\pi)^2 - (x_2-\pi)^2\right)}$	[− 10, 10]	−1	2
Elliptic function (U)	$f_{17}(x) = \sum_{i=1}^{D}\left(10^6\right)^{\frac{i-1}{D-1}} x_i^2$	[− 100, 100]	0	30
Goldstein and Price function (FD)	$f_{18}(x) = (1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14$ $x_2 + 6x_1x_2 + 3x_2^2))(30 + (2x_1 - 3x_2)^2(18 - 32x_1$	[− 2, 2]	3	2
Griewank function (M)	$f_{19} = \frac{1}{4000}\sum_{i=1}^{N} x_i^2 - \prod_{i=1}^{N}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	[− 600, 600]	0	30
Hartmann function 3 (FD)	$f_{20}(x) = -\sum_{i=1}^{4}\alpha_i \exp\left[-\sum_{j=1}^{3} A_{ij}\left(x_j - P_{ij}\right)^2\right]$	[0, 1]	−3.86278	3
Hartmann function 6 (FD)	$f_{21}(x) = -\sum_{i=1}^{4}\alpha_i \exp\left[-\sum_{j=1}^{6} A_{ij}\left(x_j - P_{ij}\right)^2\right]$	[0, 1]	−3.32237	6
Penalized 1 function (M)	$f_{22} = \frac{\pi}{n}\{10\sin(\pi y_1) + \sum_{i=1}^{n-1}(y_i - 1)2[1 + 10\sin2(\pi y_{i+1}) + (y_n - 1)2\} + \sum_{i=1}^{n} u(x_i, 10, 100, 4)$ $y_i = 1 + \frac{x_{i+1}}{4}$	[− 50, 50]	0	30
Penalized 2 function (M)	$f_{23} = 0.1\left\{(3\pi x_1) + \sum_{i=1}^{n}(x_i - 1)^2[1 + sin^2(3\pi x_i + 1)] + (x_n - 1)^2[1 + sin^2(2\pi x_n)]\right\} + \sum_{i=1}^{n} u(x_i, 5, 100, 4)$	[− 50, 50]	0	30

Table 1 (continued)

Test problems	Objective function	Search range	Optimum value	D
Rastrigin function (M)	$f_{24}(x) = 10D + \sum_{i=1}^{D}\left[x_i^2 - 10\cos(2\pi x_i)\right]$	[− 5.12, 5.12]	0	30
Scaffer function (U)	$f_{25}(x) = \left[\frac{1}{n-1}\sqrt{s_i}.\left(\sin\left(50.0 s_i^{\frac{1}{5}}\right)+1\right)\right]^2 \; s_i = \sqrt{x_i^2 + x_{i+1}^2}$	[− 100, 100]	0	30
Six Hump Camel function (FD)	$f_{26}(x) = \left(4 - 2.1x_1^2 + \frac{x_1^4}{3}\right)x_1^2 + x_1x_2 + (-4 + 4x_2^2)x_2^2$	[− 5, 5]	−1.0316	2
Weierstrass function (M)	$f_{27}(x) = \sum_{i=1}^{D}\left[\sum_{k=0}^{kmax}\left[a^k\cos\left(2\pi b^k(x_i + 0.5)\right)\right]\right] - D\sum_{k=0}^{kmax}\left[a^k\cos\left(2\pi b^k.0.5\right)\right]$; where $a = 0.5$, $b = 3$, $k\max = 20$	[− 0.5, 0.5]	0	30

4.3 Comparative study

4.3.1 Results taken by authors

The results taken by authors are presented in this subsection. Here, the proposed algorithm has been compared with the standard versions of DE [2], FA [11], FPA [13], and BA [15]. Table 2 gives the parameter description of all the algorithms, and in Table 3, the simulation results are presented. It can be seen that for f_1, only NMR has given optimum results, and no other algorithm was close to the results of NMR. In f_2, DE provides the best results, while NMR was still comparable, and with respect to other algorithms, NMR is far better. For f_3, the best achieved by each algorithm is the same, and every algorithm provides almost the same results with FPA having the best standard deviation. For f_4, FA gives a better best solution but the complication is in the worst, mean and standard deviation where the results obtained by NMR are far better. All other algorithms, in this case, provide no competitiveness. The best algorithm, in this case, is the NMR. For f_5, all algorithms give comparable results, and comparison is done in

Table 2 Parameter description of algorithms

Algorithm	Parameters	Values
FA	Number of fireflies	50
	Alpha (α)	0.5
	Beta (β)	0.2
	Gamma (γ)	1
	Maximum iterations	500
	Stopping criteria	Max iteration
DE	Population size	50
	F	0.5
	CR	0.5
	Maximum iterations	1000
	Stopping criteria	Max iteration
BA	Population size	50
	Loudness	0.5
	Pulse rate	0.5
	Maximum number of iterations	500
	Stopping criteria	Max iteration
FPA	Population size	50
	Probability switch	0.8
	Maximum iterations	500
	Stopping criteria	Max iteration
NMR	Population size	50
	Breeding probability	0.5
	λ	U (0,1)
	Maximum iterations	500
	Stopping criteria	Max iteration

Table 3 Comparison results of BA, DE, FA, FPA, and NMR

	Algorithm	Best	Worst	Mean	Standard deviation
f_1	BA	1.29E+01	1.72E+01	1.53E+01	1.04E−00
	DE	1.93E+01	2.09E+01	2.05E+01	3.00E−01
	FA	2.82E−02	1.21E−01	6.67E−02	2.09E−02
	FPA	1.10E+01	1.59E+01	1.37E+01	1.05E−00
	NMR	3.76E−12	2.70E−05	1.25E−06	5.00E−06
f_2	BA	1.20E−04	7.27E+02	2.86E+01	1.18E+02
	DE	1.29E−09	4.17E−00	1.77E−01	7.25E−01
	FA	7.90E−06	3.17E−00	1.99E−01	6.73E−01
	FPA	1.65E−01	2.51E−00	1.02E−00	6.28E−01
	NMR	8.60E−02	4.04E+01	1.02E+01	9.46E−00
f_3	BA	9.98E−01	1.71E+01	6.89E−00	4.63E−00
	DE	9.98E−01	5.92E−00	1.15E−00	7.28E−01
	FA	9.98E−01	3.96E−00	1.85E−00	7.78E−01
	FPA	9.98E−01	9.98E−01	9.98E−01	4.68E−05
	NMR	9.98E−01	8.86E−00	2.10E−00	1.57E−00
f_4	BA	8.10E+01	1.00E+05	1.51E+04	2.04E+04
	DE	1.07E+06	2.20E+06	1.64E+06	2.85E+05
	FA	7.21E−01	2.03E+01	3.75E−00	4.08E−00
	FPA	2.28E+03	2.16E+04	1.05E+04	4.61E+03
	NMR	9.86E−01	1.00E−00	9.98E−01	5.16E−05
f_5	BA	−7.4811	−3.5039	−5.7971	9.47E−01
	DE	−9.4553	−5.0448	−7.8624	1.13E−00
	FA	−9.5663	−7.1786	−8.3731	5.76E−01
	FPA	−6.6852	−5.0034	−6.0206	3.35E−01
	NMR	−8.0542	−5.5740	−6.4179	5.42E−01
f_6	BA	6.15E−01	3.21E+01	5.16E+02	7.47E+01
	DE	2.23E+03	6.87E+03	1.44E+04	3.24E+03
	FA	5.52E−01	3.36E−00	1.52E+01	2.86E−00
	FPA	1.37E+02	2.78E+02	4.83E+02	8.98E+01
	NMR	3.59E−22	2.21E−11	1.04E−09	1.48E−10
f_7	BA	1.11E−00	1.26E+01	4.51E−00	2.30E−00
	DE	1.62E+01	1.56E+02	1.07E+02	2.33E+01
	FA	7.40E−03	8.90E−02	2.77E−02	1.65E−02
	FPA	3.90E−01	2.27E−00	9.53E−01	4.29E−01
	NMR	1.14E−04	8.40E−03	1.80E−03	3.03E−04
f_8	BA	8.99E−00	1.60E+01	1.27E+01	1.73E−00
	DE	1.70E+01	2.84E+01	2.57E+01	2.28E−00
	FA	2.99E−01	7.99E−01	4.71E−01	1.29E−01
	FPA	6.04E−00	1.05E+01	7.99E−00	8.66E−01
	NMR	1.27E−09	9.99E−02	3.37E−02	4.43E−02
$f9$	BA	8.02E+03	2.65E+04	1.42E+04	3.78E+03
	DE	2.18E+04	8.02E+04	6.28E+04	1.07E+04
	FA	4.20E−03	3.25E−02	1.55E−02	6.50E−03
	FPA	2.12E+03	7.90E+03	3.82E+03	1.14E+03
	NMR	7.57E−25	1.34E−08	2.69E−10	1.89E−09
f_{10}	BA	8.52E+04	2.21E+04	1.49E+04	3.95E+03
	DE	2.96E+05	7.55E+04	6.29E+04	8.16E+03
	FA	0.00E−00	1.00E−00	2.00E−02	1.41E−01
	FPA	2.07E+03	6.54E+03	3.85E+03	9.50E+02
	NMR	0.00E−00	0.00E−00	0.00E−00	0.00E−00

Table 3 (continued)

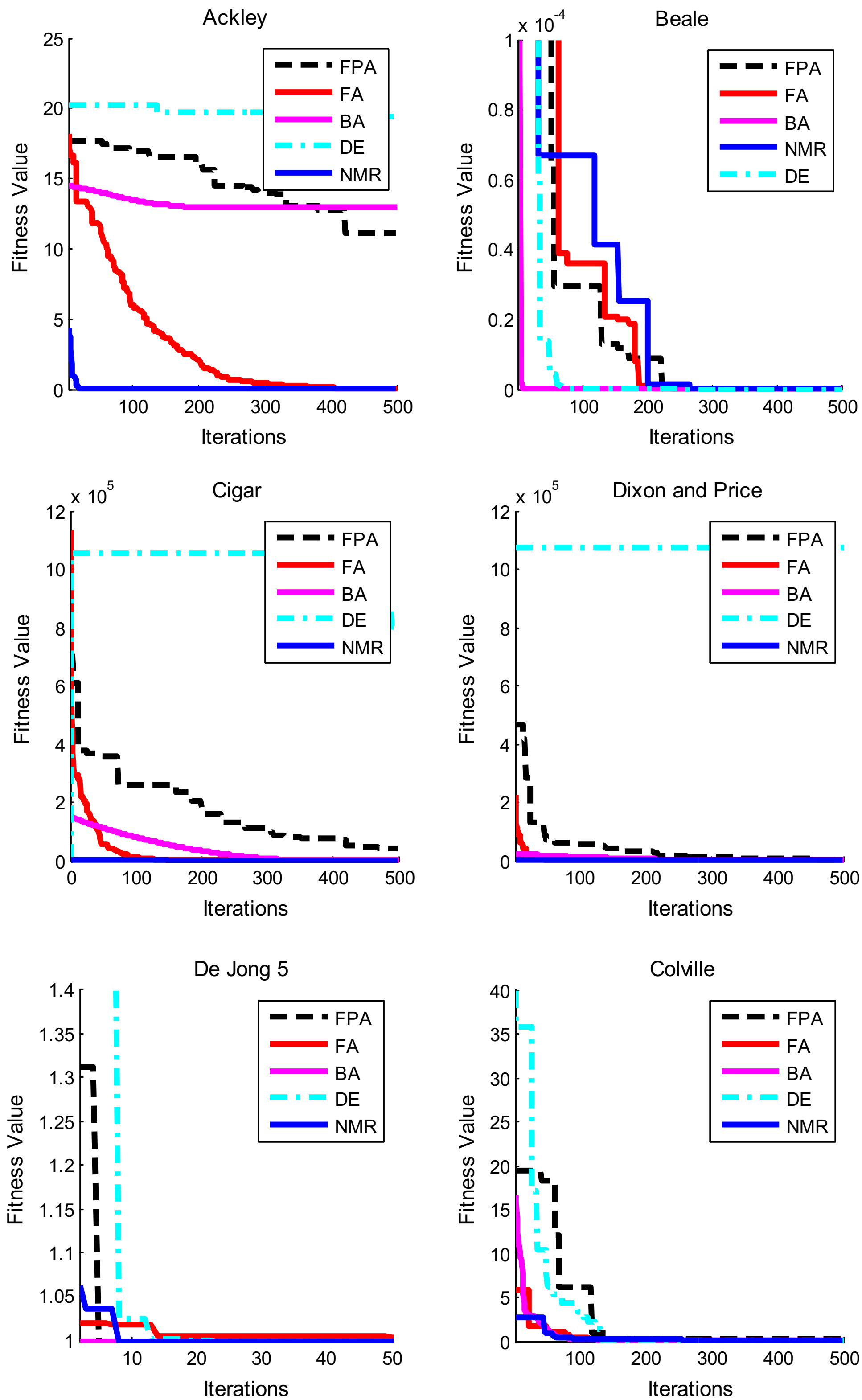
	Algorithm	Best	Worst	Mean	Standard deviation
f_{11}	BA	5.91E−09	1.41E−07	5.71E−08	3.02E−08
	DE	1.77E−01	1.43E−00	7.14E−01	3.03E−01
	FA	1.86E−08	2.61E−06	6.09E−07	5.04E−07
	FPA	8.31E−06	1.30E−03	2.51E−04	2.68E−04
	NMR	1.22E−32	2.40E−13	6.06E−15	3.49E−14
f_{12}	BA	4.16E−00	2.15E+02	5.55E+01	4.46E+01
	DE	5.91E+01	5.74E+02	1.82E+02	1.06E+02
	FA	4.92E−00	2.31E+01	1.25E+01	4.05E−00
	FPA	2.17E+01	8.23E+01	4.64E+01	1.38E+01
	NMR	2.29E−25	1.45E−08	4.50E−10	2.22E−09
f_{13}	BA	4.95E−00	7.94E+03	1.01E+03	1.30E+03
	DE	6.15E+02	3.28E+04	7.71E+03	7.54E+03
	FA	1.97E−00	1.54E+02	6.55E+01	2.64E+01
	FPA	3.13E+02	3.08E+03	1.05E+03	5.17E+02
	NMR	6.93E−24	1.05E+03	6.42E+01	1.93E+02
f_{14}	BA	1.01E−12	7.65E−01	1.03E−01	2.21E−01
	DE	0.00E−00	1.96E−12	3.92E−14	2.77E−13
	FA	9.44E−12	3.10E−09	8.63E−10	8.45E−10
	FPA	6.56E−10	1.21E−05	7.84E−07	2.30E−06
	NMR	2.73E−07	5.46E−02	4.00E−03	1.04E−02
f_{15}	BA	3.63E−01	7.76E+04	2.03E+04	2.19E+04
	DE	8.01E+05	1.82E+06	1.49E+06	2.11E+05
	FA	3.16E−00	7.34E+01	1.94E+01	1.56E+01
	FPA	4.10E+04	1.35E+05	8.84E+04	2.57E+04
	NMR	2.36E−20	5.51E−09	1.22E−10	7.80E−10
f_{16}	BA	−1.0000	−8.11E−05	−0.9800	1.41E−01
	DE	−1.0000	−1.0000	−1.0000	0.00E−00
	FA	−1.0000	−1.0000	−1.0000	3.48E−09
	FPA	−1.0000	−1.0000	−1.0000	6.26E−10
	NMR	−1.0000	−1.0000	−1.0000	4.22E−02
f_{17}	BA	1.35E+08	1.75E+09	5.12E+08	2.96E+08
	DE	1.31E+08	1.12E+09	5.86E+08	2.75E+08
	FA	9.00E+05	1.85E+07	5.17E+06	3.72E+06
	FPA	1.91E+07	8.78E+07	5.00E+07	1.67E+07
	NMR	2.93E−19	3.02E−05	6.46E−07	4.27E−06
f_{18}	BA	3	3	3	1.14E+01
	DE	3	3	3	2.71E−15
	FA	3	3	3	2.96E−08
	FPA	3	3	3	7.49E−09
	NMR	3	3	3	7.74E−01
f_{19}	BA	1.45E−01	2.17E+01	3.71E−00	4.25E−00
	DE	0.00E−00	1.92E−01	2.09E−02	5.38E−02
	FA	4.43E−08	5.54E−06	1.41E−06	1.44E−06
	FPA	2.76E−09	5.66E−05	1.03E−05	1.37E−05
	NMR	0.00E−00	1.20E−03	3.26E−04	1.80E−03
f_{20}	BA	−3.8628	−6.32E−08	−3.1546	1.19E−00
	DE	−3.8628	−3.8628	−3.8628	2.96E−15
	FA	−3.8628	−3.5663	−3.8535	4.77E−02
	FPA	−3.8628	−3.7700	−3.8493	2.04E−02

Table 3 (continued)

	Algorithm	Best	Worst	Mean	Standard deviation
	NMR	−3.8628	−3.8511	−3.8611	2.05E−03
f_{21}	BA	−3.3224	−3.2031	−3.2747	5.90E−02
	DE	−3.3224	−3.2029	−3.2241	4.54E−02
	FA	−3.3224	−3.0783	−3.2638	6.91E−02
	FPA	−3.3180	−3.2172	−3.2784	2.34E−02
	NMR	−3.3224	−3.0722	−3.1125	2.01E−02
f_{22}	BA	3.11E+04	3.96E+07	6.48E+06	9.39E+06
	DE	2.53E+08	7.38E+08	5.34E+08	1.03E+08
	FA	1.42E−04	3.90E−03	6.37E−04	5.88E−04
	FPA	3.24E+01	4.79E+05	5.02E+04	8.90E+04
	NMR	6.46E−01	1.66E−00	1.32E−00	2.70E−01
f_{23}	BA	2.94E+05	7.86E+07	2.04E+07	1.87E+07
	DE	3.38E+08	1.31E+09	1.00E+09	2.28E+08
	FA	1.00E−03	8.90E−03	3.30E−03	1.50E−03
	FPA	6.18E+04	4.38E+06	1.05E+06	9.49E+05
	NMR	2.94E−00	3.00E−00	2.99E−00	7.80E−03
f_{24}	BA	9.92E−10	4.97E−00	1.67E−00	1.53E−00
	DE	0.00E−00	1.98E−00	5.57E−01	6.08E−01
	FA	2.29E−09	2.94E−07	1.03E−07	9.11E−08
	FPA	4.81E−06	8.90E−03	7.03E−04	1.60E−03
	NMR	0.00E−00	9.95E−01	2.00E−03	1.40E−01
f_{25}	BA	4.46E−14	2.62E−01	5.44E−02	6.03E−02
	DE	0.00E−00	4.29E−02	6.80E−03	1.03E−02
	FA	2.64E−12	1.71E−02	1.60E−03	3.80E−03
	FPA	1.48E−09	1.47E−06	1.97E−07	3.20E−07
	NMR	0.00E−00	1.70E−04	3.46E−06	2.41E−05
f_{26}	BA	−1.0316	−1.0316	−1.0316	9.89E−08
	DE	−1.0316	−1.0316	−1.0316	6.72E−06
	FA	−1.0316	−1.0316	−1.0316	2.55E−09
	FPA	−1.0316	−1.0316	−1.0316	6.67E−08
	NMR	−1.0316	−1.0316	−1.0316	4.47E−09
f_{27}	BA	2.28E+01	3.83E+01	3.14E+01	3.23E−00
	DE	1.92E+01	3.93E+01	2.86E+01	4.77E−00
	FA	8.10E−00	2.15E+01	1.60E+01	2.80E−00
	FPA	3.16E+01	3.63E+01	3.36E+01	1.06E−00
	NMR	6.39E−13	5.50E−03	3.02E−04	8.45E−08

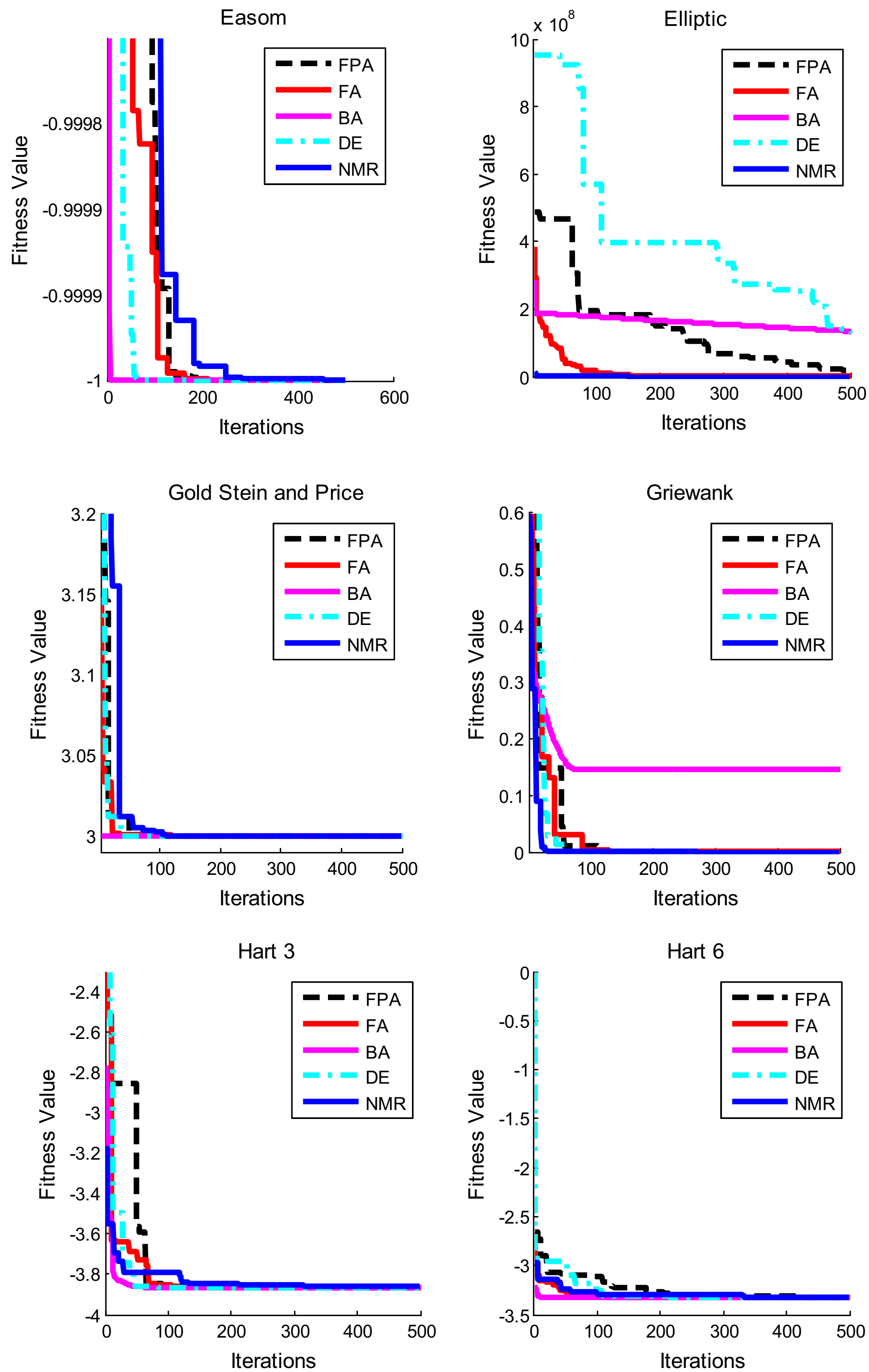
terms of standard deviation. It can be seen that FPA and NMR are better than others. For f_6, f_7, f_8, and f_9, NMR is able to achieve the minimum best solution, even in terms of worst, mean and standard deviation, the results are much better than the others. For f_{10}, NMR and FA provide exact global optimum results, while others were not able to achieve any good results. Here, the FA was able to attain only the global best solution, whereas NMR provided exact zero optimum for all the results. For f_{11}, NMR is the best, while all others are found to be highly competitive. For f_{12}, and f_{13}, NMR is the best, while others get stuck in some local optima. For f_{14}, BA, FA, FPA, and NMR are highly competitive, but here DE gives the best results. For f_{15}, NMR is found to be better among all in terms of best, worst, mean as well as standard deviation. Functions f_{16}, f_{18}, f_{20}, f_{21}, and f_{26} all are fixed dimension, and results are judged on the basis of standard deviation. Here, the algorithm with a minimum standard deviation is the best. Here for f_{16}, f_{18}, f_{20}, DE provide the best standard deviation, for f_{21}, all are comparable but NMR has the minimum among all and for f_{26}, FA and NMR show almost same results, while others are comparable. For f_{17} f_{19}, and f_{22}, NMR is

Fig. 5 Convergence profiles



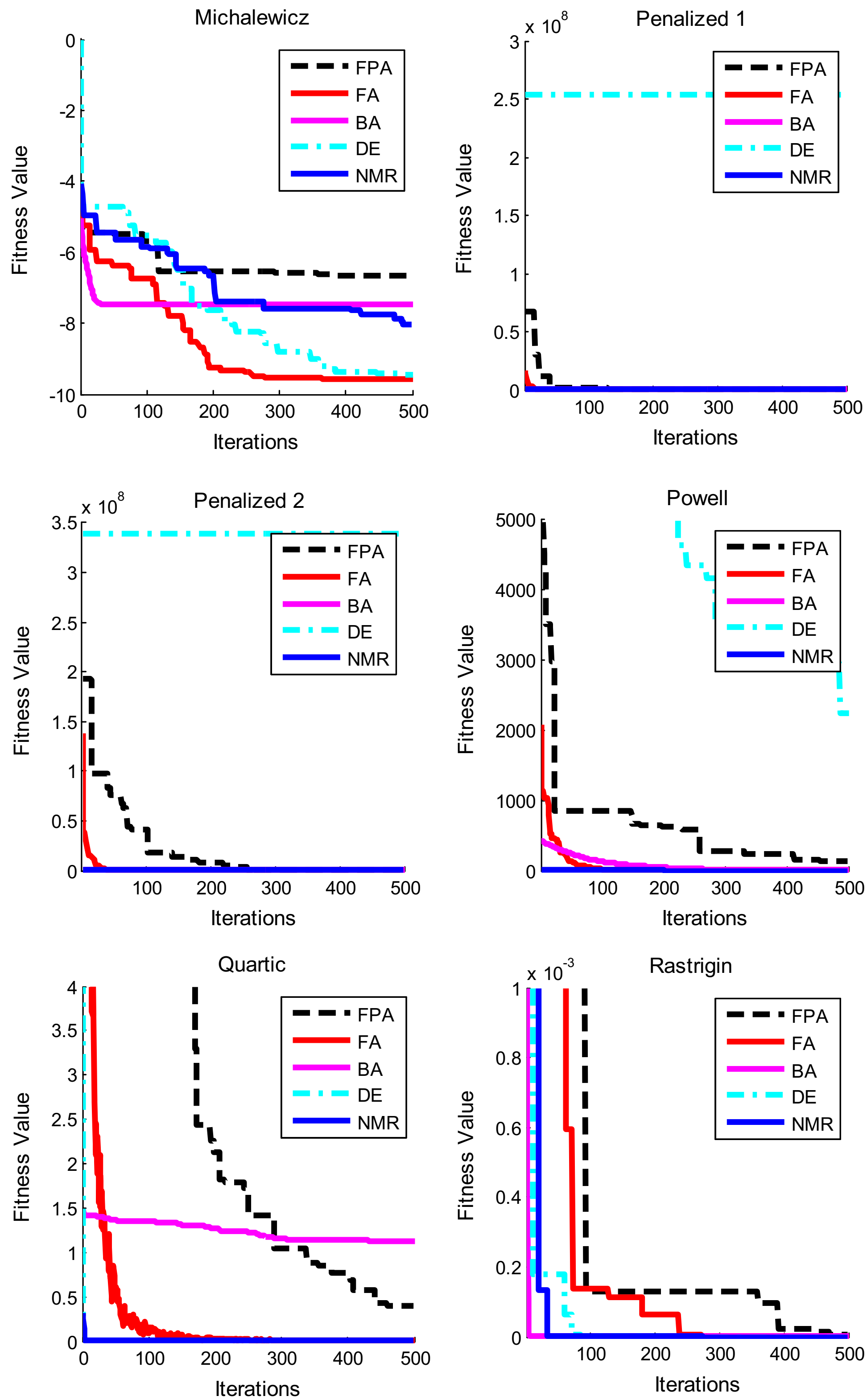
better in every aspect and for f_{23} FA is comparable to NMR but still, NMR is better. For f_{24} and f_{25}, DE and NMR provide the exact zero global optimum. Here in terms of worst, mean and standard deviation, NMR is better. For f_{27}, NMR gives the best results. Overall, DE is found to be better for five functions, FPA for two functions, FA for one, and for rest of the nineteen functions, the proposed NMR provides better results.

Fig. 5 continued



The convergence plots in Fig. 5 also show the performance of NMR with respect to BA, FA, DE, and FPA. According to Berg et al. [53], for an algorithm to converge to a point in search space, search agents should change abruptly at the start or initial stages of optimization and converge gradually as the iterations proceed. The convergence curve for each algorithm is drawn by using the best fitness values over 500 iterations and 50 runs. It is shown in Fig. 5 that for most of the functions, NMR is found to

Fig. 5 continued



follow the pattern proposed by Berg et al. Further, the curves also show that NMR algorithm is superior to others.

Wilcoxon's rank-sum test [54] is performed to significantly test the performance of NMR with other popular algorithms. It is a nonparametric test, involving the design of two samples and is analogous to paired t test. So, it is a pairwise test used to identify significant differences between the two algorithms. The test gives a p-value as the

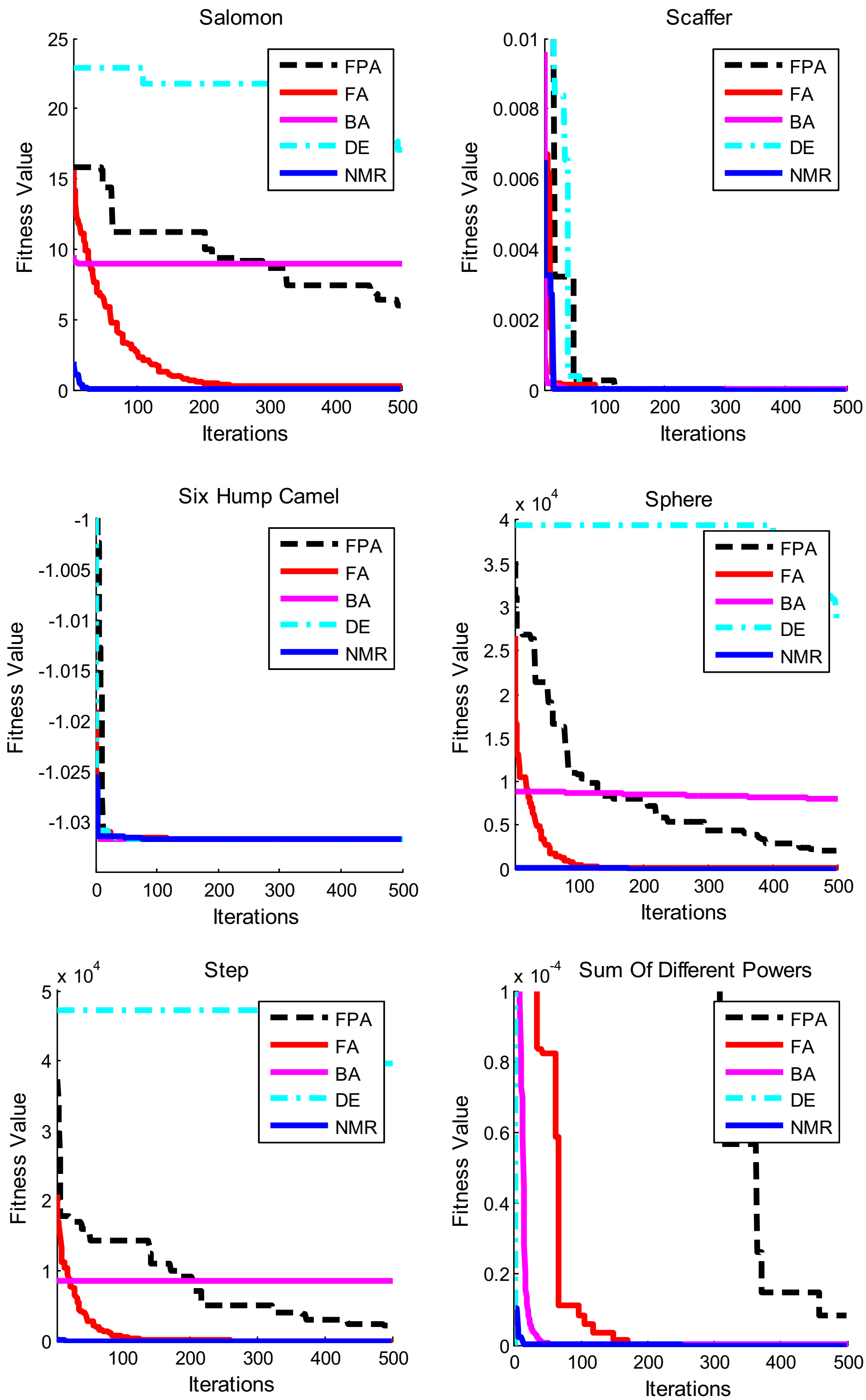
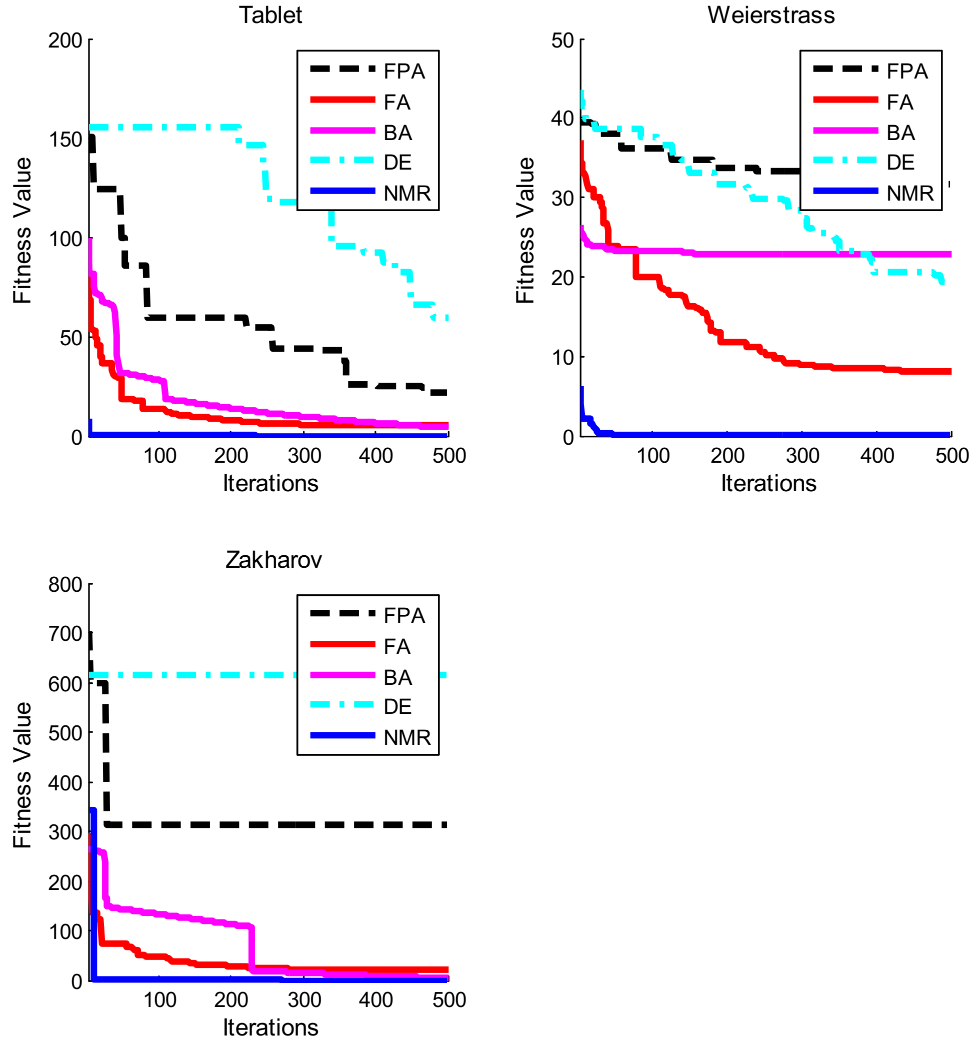


Fig. 5 continued

end result, and this p value determines the significance level of two algorithms under test. An algorithm is said to be statistically significant if the p value is < 0.05. In Table 4, the places where NA is written correspond to the best algorithm with respect to others. Again, from Table 4, it can be seen that DE gives best results for five functions, FA for one, FPA for two, BA for none, and for rest of the functions, NMR is more statistically significant.

Fig. 5 continued



4.3.2 Results from the literature

For comparing the results of the literature, eleven test functions have been used. The algorithms used for comparison are whale optimization algorithm (WOA) [55], GWO [56], PSO [10], gravitational search algorithm (GSA) [57], and fast evolutionary programming (FEP) [58]. The results of WOA, PSO, and FEP have been taken from [55]. The evaluation criteria for these set of results are in terms of the maximum number of function evaluations, and the maximum number of function evaluations is given by population size multiplied by the total number of iterations. For NMR, the population size is taken as 30, and the maximum number of iterations is 500. All other algorithms use the same set of population size and iterations as reported in the literature.

In Table 5, the simulation results in comparison with the literature are presented. Here, all the functions are taken from Table 1. The functions used are f_1, f_3, f_9, f_{10}, f_{19}, f_{20}, f_{21}, f_{22}, f_{23}, f_{24}, and f_{26}. It can be seen that for f_1, NMR is far better than WOA, PSO, GSA, and FEP. Here, only GWO is found to be the best algorithm. For f_3, the best achieved by each algorithm is similar, and every algorithm provides almost the same results with FEP having the best standard deviation. For f_9, NMR is better than PSO and FEP and given comparable results with respect to others. For f_{10}, NMR gives exact global optimum results, while others were not able to achieve any good results. Here, the standard deviation is also found to be exactly zero. For f_{19}, WOA is superior to NMR, while all other algorithms, NMR is found to be highly competitive. Functions f_{20}, f_{21}, and f_{26} all are fixed dimension, and the results are judged on the basis of standard deviation. In these functions, GWO is found to provide the worst standard deviation, while all others are comparable. Also, the standard deviation presented by GWO is not exactly the standard deviation and maybe the results reported by the author are vague. These results have been marked with an asterisk (*) sign. Here, GSA is found to provide the best results. For f_{22} and f_{23}, FEP is better in every aspect, and among others, NMR is

Table 4 Wilcoxon rank-sum test comparison of all algorithms

Function	BA	DE	FA	FPA	NMR
f_1	7.06E−18	7.06E−18	7.06E−18	7.06E−18	NA
f_2	3.13E−20	NA	3.13E−20	3.13E−20	3.31E−20
f_3	3.13E−20	3.13E−20	3.13E−20	NA	3.31E−20
f_4	3.13E−20	3.13E−20	3.13E−20	3.13E−20	NA
f_5	7.06E−18	7.06E−18	7.06E−18	7.06E−18	NA
f_6	3.31E−20	3.31E−20	3.31E−20	NA	3.31E−20
f_7	3.31E−20	3.31E−20	3.31E−20	3.31E−20	NA
f_8	7.06E−18	7.06E−18	7.06E−18	7.06E−18	NA
f_9	7.06E−18	7.06E−18	7.06E−18	7.06E−18	NA
f_{10}	3.31E−20	3.31E−20	3.31E−20	3.31E−20	NA
f_{11}	3.31E−20	3.31E−20	3.31E−20	3.31E−20	NA
f_{12}	3.31E−20	3.31E−20	3.31E−20	3.31E−20	NA
f_{13}	3.31E−20	3.31E−20	3.31E−20	3.31E−20	NA
f_{14}	0.0123	NA	0.0123	0.0123	0.0123
f_{15}	3.31E−20	3.31E−20	3.31E−20	3.31E−20	NA
f_{16}	3.31E−20	NA	3.31E−20	3.31E−20	3.31E−20
f_{17}	7.06E−18	7.06E−18	7.06E−18	7.06E−18	NA
f_{18}	9.45E−19	NA	9.45E−19	9.45E−19	9.45E−19
f_{19}	5.87E−04	5.87E−04	5.87E−04	5.87E−04	NA
f_{20}	1.91E−18	NA	1.91E−18	1.91E−18	1.91E−18
f_{21}	5.27E−05	5.27E−05	5.27E−05	5.27E−05	NA
f_{22}	7.06E−18	7.06E−18	7.06E−18	7.06E−18	NA
f_{23}	7.06E−18	7.06E−18	7.06E−18	7.06E−18	NA
f_{24}	3.85E−07	3.85E−07	3.85E−07	3.85E−07	NA
f_{25}	3.31E−20	3.31E−20	3.31E−20	3.31E−20	NA
f_{26}	7.06E−18	7.06E−18	NA	7.06E−18	7.06E−18
f_{27}	3.31E−20	3.31E−20	3.31E−20	3.31E−20	NA

found to be comparable. For f_{24}, NMR is found to provide the best solutions. Overall, we can say that NMR is highly competitive even when compared to other recently proposed algorithms.

4.4 Discussion of results

Exploration and exploitation are the two necessary phenomena of evolutionary computing. These two processes decide whether an algorithm is performing better or not. Exploration means global search and is meant for exploring the whole of the search space, whereas exploitation is concerned with the local search or simply intensive search to achieve global results. It has been found in the literature that both these phenomena should be balanced to achieve better results. However, it is still to be explored that to what extent local and global search are to be performed [59].

In NMR algorithm, exploration and exploitation are balanced by the worker and breeder phases. Here, workers because of their large population size perform exploration. The workers continuously update themselves in order to become a part of the breeder's group and in turn tend to explore different regions of the search space. The breeders belong to a population with higher fitness than the workers and perform more intensive exploitation as they tend to improve their fitness in order to mate with the queen. The breeders tend to follow the best-known breeder, and hence take small steps to improve the good solutions. Moreover, if a breeder does not improve its fitness, it is pushed into the worker's group, so it avoids solutions getting stuck in local minima and of stagnation of population of breeders. Further, exploration and exploitation of NMR have been tested by applying it to unimodal and multimodal functions. Here, unimodal functions have only one global optimum, and hence help to evaluate the exploitation tendencies of an algorithm. For multimodal functions, there are more than one local optimum, and their number increases exponentially with population size. So, these functions are capable of evaluating the explorative tendencies. From the results in Tables 3 and 4, it can be seen that the NMR algorithm is able to outperform other algorithms and is mostly at the first or second place in the

Table 5 Result comparison of WOA, GWO, PSO, GSA, FEP, and NMR

Function		WOA	GWO	PSO	GSA	FEP	NMR
f_1	Mean	7.40E−00	1.06E−13	2.76E−01	6.20E−02	1.80E−02	2.69E−05
	Std	9.89E−00	7.78E−02	5.09E−01	2.36E−01	2.10E−03	6.94E−05
f_3	Mean	2.11E−00	4.04E−00	3.62E−00	5.85E−00	1.22E−00	2.42E−00
	Std	2.94E−00	4.25E−00	2.56E−00	3.83E−00	5.60E−01	1.79E−00
f_9	Mean	1.41E−30	6.28E−28	1.36E−03	2.53E−16	5.70E−04	9.11E−09
	Std	4.91E−30	6.34E−05	2.02E−04	9.67E−17	1.30E−04	4.86E−08
f_{10}	Mean	3.11E−00	8.16E−01	1.02E−03	2.50E−16	0.00E−00	0.00E−00
	Std	5.32E−01	1.26E−03	8.28E−05	1.74E−16	0.00E−00	0.00E−00
f_{19}	Mean	2.89E−04	4.48E−03	9.21E−03	2.77E+01	1.60E−02	2.70E−03
	Std	1.58E−03	6.65E−03	7.72E−03	5.04E−00	2.20E−01	1.28E−02
f_{20}	Mean	−3.85616	−3.86263	−3.86278	−3.86278	−3.86000	−3.85060
	Std	2.70E−03	−3.86278*	2.58E−15	2.29E−15	1.40E−05	3.99E−01
f_{21}	Mean	−2.98105	−3.28865	−3.26634	−3.31778	−3.27000	−3.31760
	Std	3.76E−01	−3.25065*	6.05E−02	2.30E−02	5.90E−02	3.84E−01
f_{22}	Mean	3.39E−01	5.34E−02	6.91E−03	1.79E−00	9.20E−06	1.28E−00
	Std	2.14E−01	2.07E−02	2.63E−02	9.51E−01	3.60E−06	3.20E−01
f_{23}	Mean	1.88E−00	6.54E−01	6.67E−03	8.89E−00	1.60E−04	2.90E−00
	Std	2.66E−01	4.47E−03	8.90E−03	7.12E−00	7.30E−05	1.00E−03
f_{24}	Mean	0.00E−00	3.10E−01	4.67E+01	2.59E+01	4.60E−02	9.00E−03
	Std	0.00E−00	4.73E+01	1.16E+01	7.47E−00	1.20E−02	4.90E−02
f_{26}	Mean	−1.0316	−1.0316	−1.0316	−1.0316	−1.0316	−1.0316
	Std	4.20E−07	−1.03163*	6.25E−16	4.88E−16	4.90E−07	4.30E−03

*Std in the table stands for Standard deviation values

majority of the test functions. This is due to the integrated mechanism of exploration and exploitation in NMR.

In this article, the parameters such as population size, dimension, breeding probability, and mating factor are not exploited because the article is concerned with the proposal of a new optimization technique and so the exploitation of parameters is left for future works.

5 Conclusion

A new swarm intelligent optimization algorithm based on the mating behavior of naked mole rats has been presented in this paper. The algorithm is inspired by the naked moles and divides the population into two groups, namely workers and breeders. Workers are responsible for exploring the search space, and breeders tend to exploit good solutions. The algorithm has been tested on 27 benchmark functions to verify its explorative and exploitative behaviors, avoidance of local minima, and convergence speed. Its performance has been compared with popular and state-of-the-art algorithms. It was found that NMR algorithm shows good exploitation and exploration for unimodal and multimodal functions, respectively. Moreover, its performance was better than FA, DE, BA, FPA, GWO, WOA, PSO, GSA, and FEP algorithms for most of the benchmark functions in terms of solution quality and convergence speed. The Wilcoxon rank-sum test also proved that NMR is able to give optimum solutions consistently with respect to other algorithms.

In the future, this algorithm may be tested for real-world problems and see how it performs. Extending the algorithm to the binary version can be used in an electroencephalogram (EEG). In EEG, signals are measured by placing sensors in different positions on a person's head. A binary version of NMR can help to reduce the total number of sensors required for the operation while maintaining the accuracy of the system. The binary version can also be used for thinning of antennas and placement of nodes in wireless sensor networks. Moreover, its multi-objective version can be developed for optimization problems having many conflicting objectives.

Acknowledgements Rohit Salgotra acknowledges the support of INSPIRE Fellowship (IF-160215) by Directorate of Science and Technology, Govt. of India.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

Human and animal rights This article does not contain any studies with human or animal subjects performed by any of the authors.

Informed consent All procedures followed were in accordance with the ethical standards of the responsible committee on human experimentation (institutional and national) and with the Helsinki Declaration of 1975, as revised in 2008 (5).

References

1. Holland JH (1992) Genetic algorithms. Sci Am 267(1):66–72
2. Storn R, Price K (1997) Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. J Global Optim 11(4):341–359
3. Rechenberg I (1978) Evolutionsstrategien. In: Simulationsmethoden in der Medizin und Biologie. Springer Berlin Heidelberg, pp 83–114
4. Simon D (2008) Biogeography-based optimization. IEEE Trans Evol Comput 12(6):702–713
5. Koza JR (1992) Genetic programming: on the programming of computers by means of natural selection, vol 1. MIT Press, Cambridge
6. Mirjalili S (2015) The ant lion optimizer. Adv Eng Softw 83:80–98
7. Dasgupta D, Michalewicz Z (eds) (2013) Evolutionary algorithms in engineering applications. Springer, Berlin
8. Beni G, Wang J (1993) Swarm intelligence in cellular robotic systems. In: Robots and biological systems: towards a new bionics? Springer Berlin Heidelberg, pp 703–712
9. Karaboga D (2005) An idea based on honey bee swarm for numerical optimization, vol 200. Technical report-tr06, Erciyes University, Engineering Faculty, Computer Engineering Department
10. Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of the 1995 IEEE international conference on neural networks, pp 1942–1948
11. Yang XS (2010) Firefly algorithm, stochastic test functions, and design optimisation. International Journal of Bio-Inspired Computation 2(2):78–84
12. Passino KM (2002) Biomimicry of bacterial foraging for distributed optimization and control. IEEE control systems 22(3):52–67
13. Yang XS (2012) Flower pollination algorithm for global optimization. In: International conference on unconventional computing and natural computation. Springer Berlin Heidelberg, pp 240–249
14. Singh U, Salgotra R (2018) Synthesis of linear antenna array using flower pollination algorithm. Neural Comput Appl 29(2):435–445
15. Yang XS (2010) A new metaheuristic bat-inspired algorithm. In: Nature inspired cooperative strategies for optimization (NICSO 2010). Springer Berlin Heidelberg, pp 65–74
16. Faris H, Mirjalili S, Aljarah I, Mafarja M, Heidari AA (2020) Salp swarm algorithm: theory, literature review, and application in extreme learning machines. In: Nature-inspired optimizers. Springer, Cham, pp 185–199
17. Khalilpourazari S, Khalilpourazary S (2018) Optimization of time, cost and surface roughness in grinding process using a robust multi-objective dragonfly algorithm. Neural Comput Appl 1–12
18. Salgotra R, Singh U, Saha S (2018) New cuckoo search algorithms with enhanced exploration and exploitation properties. Expert Syst Appl 95:384–420
19. El Aziz MA, Hassanien AE (2018) Modified cuckoo search algorithm with rough sets for feature selection. Neural Comput Appl 29(4):925–934
20. Armaghani DJ, Hasanipanah M, Mahdiyar A, Majid MZA, Amnieh HB, Tahir MM (2018) Airblast prediction through a hybrid genetic algorithm-ANN model. Neural Comput Appl 29(9):619–629
21. Liu A, Li P, Sun W, Deng X, Li W, Zhao Y, Liu B (2019). Prediction of mechanical properties of micro-alloyed steels via neural networks learned by water wave optimization. Neural Comput Appl 1–16
22. Aljarah I, Faris H, Mirjalili S (2018) Optimizing connection weights in neural networks using the whale optimization algorithm. Soft Comput 22(1):1–15
23. Kaur K, Singh U, Salgotra R (2018) An enhanced moth flame optimization. Neural Comput Appl 1–35
24. Bonabeau E, Dorigo M, Theraulaz G (1999) Swarm intelligence: from natural to artificial systems (No. 1). Oxford University Press, New York
25. Jeanne RL (1986) The evolution of the organization of work in social insects. Monitore Zoologico Italiano-Italian Journal of Zoology 20(2):119–133
26. Oster GF, Wilson EO (1978) Caste and ecology in the social insects. Princeton University Press, Princeton
27. Wolpert DH, Macready WG (1997) No free lunch theorems for optimization. IEEE Trans Evol Comput 1(1):67–82
28. Karaboga D, Basturk B (2008) On the performance of artificial bee colony (ABC) algorithm. Appl Soft Comput 8(1):687–697
29. Lampinen J, Zelinka I (2000) On stagnation of the differential evolution algorithm. In: Proceedings of MENDEL, pp 76–83
30. Shi Y, Eberhart RC (1998) Parameter selection in particle swarm optimization. In: International conference on evolutionary programming. Springer Berlin Heidelberg, pp 591–600
31. Blouin SF, Blouin M (1988) Inbreeding avoidance behaviors. Trends Ecol Evol 3(9):230–233
32. Niu P, Niu S, Chang L (2019) The defect of the Grey Wolf optimization algorithm and its verification method. Knowl Based Syst
33. Williams GC (1975) Sex and evolution (No. 8). Princeton University Press, Princeton
34. Hamilton WD (1993) Inbreeding in Egypt and in this book: a childish perspective. The natural history of Inbreeding and outbreeding, pp 429–450
35. Bateson P (1983) Optimal outbreeding. Mate Choice 257:277
36. Thornhill NW (1993) The natural history of inbreeding and outbreeding: theoretical and empirical perspectives. University of Chicago Press, Chicago
37. Smith RH (1979) On selection for inbreeding in polygynous animals. Heredity 43(2):205–211
38. Parker GA (1979) Sexual selection and sexual conflict. In: Blum MS, Blum NA (eds) Sexual selection and reproductive competition in insects, pp 123–163
39. Ciszek D (2000) New colony formation in the “highly inbred” eusocial naked mole-rat: outbreeding is preferred. Behav Ecol 11(1):1–6
40. Buffenstein R, Jarvis JU, Opperman LA, Cavaleros M, Ross FP, Pettifor JM (1994) Subterranean mole-rats naturally have an impoverished calciol status, yet synthesize calciol metabolites and calbindins. Eur J Endocrinol 130(4):402–409
41. Sheffield SR, Sawicka-Kapusta K, Cohen JB, Rattner BA (2001) Rodentia and Lagomorpha
42. Wilson DE, Reeder DM (eds) (2005) Mammal species of the world: a taxonomic and geographic reference. JHU Press, Baltimore

43. Deuve JL, Bennett NC, Britton-Davidian J, Robinson TJ (2008) Chromosomal phylogeny and evolution of the African mole-rats (Bathyergidae). Chromosome Res 16(1):57–74
44. Brett RA (1991) The population structure of naked mole-rat colonies. The biology of the naked mole-rat, 97
45. Sherman PW, Jarvis JUM, Alexander RD (eds) (1991) The biology of inbreeding and outbreeding. University of Chicago Press, Chicago
46. Edrey YH, Hanes M, Pinto M, Mele J, Buffenstein R (2011) Successful aging and sustained good health in the naked mole rat: a long-lived mammalian model for biogerontology and biomedical research. ILAR J 52(1):41–53
47. O'Riain MJ, Jarvis JUM, Alexander R, Buffenstein R, Peeters C (2000) Morphological castes in a vertebrate. Proc Natl Acad Sci 97(24):13194–13197
48. Crish SD, Dengler-Crish CM, Catania KC (2006) Central visual system of the naked mole-rat (Heterocephalus glaber). Anat Rec A Discov Mol Cell Evol Biol 288(2):205–212
49. Faulkes CG, Abbott DH, Jarvis JUM, Sherriff FE (1990) LH responses of female naked mole-rats, Heterocephalus glaber, to single and multiple doses of exogenous GnRH. J Reprod Fertil 89(1):317–323
50. Clarke FM, Faulkes CG (1998) Hormonal and behavioral correlates of male dominance and reproductive status in captive colonies of the naked mole–rat, Heterocephalus glaber. Proceedings of the Royal Society of London B: Biological Sciences 265(1404):1391–1399
51. Alexander RD, Noonan KM, Crespi BJ (1991) The evolution of eusociality. The biology of the naked mole-rat 3:44
52. Jarvis JU (1981) Eusociality in a mammal: cooperative breeding in naked mole-rat colonies. Science 212(4494):571–573
53. Van Den Bergh F, Engelbrecht AP (2006) A study of particle swarm optimization particle trajectories. Inf Sci 176(8):937–971
54. Derrac J, García S, Molina D, Herrera F (2011) A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm and Evolutionary Computation 1(1):3–18
55. Mirjalili S, Lewis A (2016) The whale optimization algorithm. Adv Eng Softw 95:51–67
56. Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. Adv Eng Softw 69:46–61
57. Rashedi E, Nezamabadi-Pour H, Saryazdi S (2009) GSA: a gravitational search algorithm. Inf Sci 179(13):2232–2248
58. Yao X, Liu Y, Lin G (1999) Evolutionary programming made faster. IEEE Trans Evol Comput 3(2):82–102
59. Črepinšek M, Liu SH, Mernik M (2013) Exploration and exploitation in evolutionary algorithms: a survey. ACM Computing Surveys (CSUR) 45(3):35