

A Spatially Informative Optic Flow Model of Bee Colony With Saccadic Flight Strategy for Global Optimization

Swagatam Das, *Senior Member, IEEE*, Subhodip Biswas, Bijaya K. Panigrahi, *Senior Member, IEEE*, Souvik Kundu, and Debabrota Basu

Abstract—This paper presents a novel search metaheuristic inspired from the physical interpretation of the optic flow of information in honeybees about the spatial surroundings that help them orient themselves and navigate through search space while foraging. The interpreted behavior combined with the minimal foraging is simulated by the artificial bee colony algorithm to develop a robust search technique that exhibits elevated performance in multidimensional objective space. Through detailed experimental study and rigorous analysis, we highlight the statistical superiority enjoyed by our algorithm over a wide variety of functions as compared to some highly competitive state-of-the-art methods.

Index Terms—Artificial bee colony algorithm, foraging, global optimization, locomotion, metaheuristics, optic flow model, saccadic flight, swarm.

I. INTRODUCTION

SCIENTIFIC communities believe that honeybee foragers are quick in associative learning [1], [2]. They can save energy during their search for food by not repeating movements to already visited food sites, unless they get sufficiently rewarded by the presence of reasonable food content. By drawing an analogy of this minimal foraging model of honeybees, we can design metaheuristics that will efficiently utilize computational resources. One of the foremost steps that was taken in this direction was that by Sato and Hagiwara [3] who proposed the first algorithm based on bees foraging called the Bee System (BS). Later on, more in-depth studies

Manuscript received May 4, 2013; revised September 30, 2013; accepted December 14, 2013. Date of publication January 30, 2014; date of current version September 12, 2014. This paper was recommended by Associate Editor Y. Jin.

S. Das is with the Electronics and Communication Sciences Unit, Indian Statistical Institute, Kolkata 700108, India (e-mail: swagatam.das@isical.ac.in).

S. Biswas, S. Kundu, and D. Basu are with the Department of Electronics and Telecommunication Engineering, Jadavpur University, Kolkata 700032, India (e-mail: sub17was@gmail.com; sk210892@gmail.com; basudebabrota29@gmail.com).

B. K. Panigrahi is with the Department of Electrical Engineering Unit, Indian Institute of Technology Delhi, Delhi 110016, India (e-mail: bkpanigrahi@ee.iitd.ac.in).

This paper has supplementary downloadable multimedia material available at <http://ieeexplore.ieee.org> provided by the authors. This include a PDF file with experimental details, convergence analyses, and survey materials relevant to the original paper. These materials complement the study presented in the paper within the page restrictions of this journal. The material is 350 KB in size.

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCYB.2014.2298916

were conducted on the dance and communication, collective decision, task allocation, mating, nest site selection, marriage, foraging, floral and pheromone laying, and navigation behaviors of the bees swarm. These led to the development of algorithms based on bee swarm intelligence (SI) [4], for example, virtual bee, BeeAdHoc, the marriage in honeybees, the BeeHive, bee colony optimization, etc. The turning point in the recognition of the bee SI algorithms came with the proposal of the Artificial Bee Colony (ABC) algorithm [5] by simulating the foraging model of bees. It accounts for more than half of the research work involving bee colony-based metaheuristics. Interested readers can refer to the survey by Karaboga *et al.* [4] to see the works involving ABC algorithm.

However, ABC too has its own drawbacks. Biological findings [1], [2], [6] suggest that the ability of honeybees to forage is dependent on their locomotion ability in the search space. This, in turn, depends on certain biotic factors, primarily on the flow of visual information from the surroundings. Locomotion is the basic life-supporting mechanism that enables a living being to travel in its environment, escape from potential danger, and most importantly to collect food. As a matter of fact, most of our behaviors are highly dependent on our ability to see, and most organisms are blessed with visual control movement maneuvers (saccadic flight) that help them adjust their motion to avert collision, escape encounter with potential threat, instantiate reflexive movement, and so on.

The classical ABC [5] treats a set of search points as food sources and proceeds by changing the position of the search points just like a forager changing its position to find better food sources. But this approach depends only on the position of food sources and their fitness value. In contrast to the biological factors that form the actual basis of decision making in the foragers' food site selection, basic ABC implementation neglects most of such criteria. One such important factor is the ratio of fitness value T (nectar content of food sources) to the distance E (traveled by foragers). In addition, the movement of the foragers is guided by certain geo-spatial factors, which are neglected by the ABC algorithm. Looking at the broader picture, we see that there has been scarcely any attempt to unify the behavioral characteristics of the bee SI under one roof. Although it is practically not possible to replicate all the characteristics, a subset of these characteristics, such as communication, collective decision, task allocation, nest site

selection, foraging, and navigation behaviors of the swarm, when properly implemented in a synchronous manner, can lead to the devisal of a more efficient means of problem solving as compared to its predecessors.

Motivated by these findings, we propose an optimization technique called spatially informative perturbation-based ABC algorithm with saccadic flight (*SiPABC_Sf*), based on the optic flow [6] of geo-spatial information about the surrounding location in the foragers. The *SiPABC_Sf* synergizes the basics of the bee foraging model and associative learning with the locomotion-based behaviors that have their sources in the optic flow of dynamic data relative to its position in the environment. Note that the source of the synchronism implemented by our algorithm keeps true to its starting point—*vision*, the ability to see. *SiPABC_Sf* makes use of locomotion along intelligent directions in the search space via saccadic flight strategy [7], which is inspired from the sharp maneuvers of the head and the body employed by honeybees to familiarize themselves with the spatial layout of the surroundings. The steps of the algorithm can be broadly classified into two simple stages. The first stage involves the selection of guided locomotion modes that diversify the local search process based on color learning in the foragers. It is simulated by an initial exposure to a specific color corresponding to a locomotion mode. Then, it adapts itself to the suitable mode through associative learning implemented mathematically. The second stage is involved with the processing of neighborhood information. The saccadic flight strategy is based on an evaluation of strategically effective sites for search as modeled by its two components: 1) tradeoff function and 2) correlative neurological model based on the combined feedback arising from the centering response and the directionally selective motion response.

The rest of the paper is organized as follows. Section II discusses the background details of the bee colony to assist in understanding their organization, mode of communication, and related studies on their observed behavior. Section III provides an overview of the classical ABC algorithm. Section IV presents the *SiPABC_Sf* algorithm in sufficient detail, outlining the functional philosophy of the integrated approaches undertaken. Section V discusses the experimental results of comparing the performance of our algorithm. Finally, Section VI concludes the paper unraveling some important future avenues of research. To gain better understanding, interested readers can refer to the supplementary file.

II. BACKGROUND DETAILS: FORAGING, LEARNING, AND LOCOMOTIVE BEHAVIOR IN HONEY BEES

An insect colony, although made up of many identical members, can function cooperatively in a synchronous manner as a superorganism [8]. The degree of their synchronism, cooperation, task force division, and dynamic interaction go a long way in ensuring their survival just like the cells in our body.

A. Foraging and Communication

- 1) *Food source*: The initial search for food begins by seeking trial food sources, which are gradually discarded

during the ongoing search for fitter ones. Depending on external parameters, such as distance, nectar concentration, and ease of extraction, the fitness of a food source is judged. The basic idea is to memorize the location of the food source if it is found to be adequately rewarding in nature.

- 2) *Unemployed Foragers*: The foraging action begins with the bees having no prior knowledge about the food site. A bee is categorized as a scout bee if it searches or a recruit who attends to a bio-communication mode called the waggle dance and goes to the site using the information conveyed to it.
- 3) *Employed Foragers*: The unemployed foragers, on encountering a food source, are employed at it and start performing exploitation. While loading nectar, a forager memorizes the information like distance and location. It returns back to the nest and deposits the collected nectar. Following the unloading process, it can choose to further exploit the site, abandon the source found if it is exhausted, or recruit taskforce by conveying information about the food source to the fellow foragers via waggle dance.

Communication [9] is of absolute necessity to transmit information about resources. Waggle dance [10] is one such information-sharing mechanism through which the foragers can share the details regarding the direction and distance of the nectar site from the hive with the dance duration being directly proportional to the distance. The to-and-fro commute between the food source and the hive can be executed by the bees successfully by map-oriented navigation, as pointed out in [11] regarding spatial memory of the path traversed.

B. Color Learning in Bees

Austrian zoologist Karl von Frisch [12] was the first person to have vouched for the presence of color vision in honeybees after an extensive experimental analysis. Later on, Menzel carried out the work started by von Frisch and conducted tests to understand the color learning and visual memory. It was affirmed that the bees can retain past information about the color of the source from which they got rewarded for a span of a few days after the initial exposure. He also concluded about the preferential ease of color learning in bees with regard to the violet color since most of the flowers are identical to it. The visual memory has been put to use by demarcating the colors as locomotive indicators associated with a given mode of locomotion. The interlinking between color learning and visual memory has been utilized in our algorithm design.

C. Correlative Representation of Elementary Motion Detection

Optic flow of spatial information takes place when the bees traverse through the environment during locomotion and it is continually analyzed by the parallel motion detectors associated with the neural system of the bees. The data assimilating from the vast visual field are combined and analyzed in the presence of photoreceptors. The primary purpose of the motion detectors is to detect local motion information during the

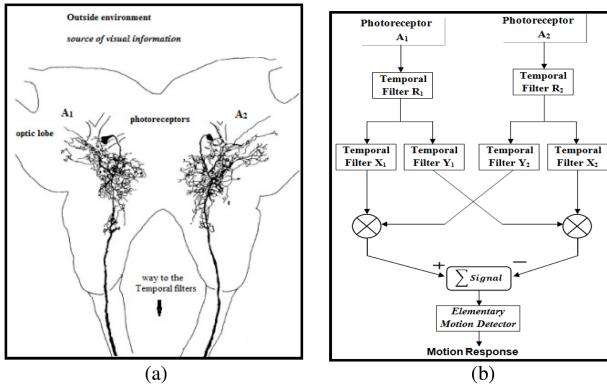


Fig. 1. Representation of the correlative optic model for motion detection. (a) Biological diagram. (b) Schematic diagram.

propagation and generate a corrective yaw that compensates for the unwanted deviations that the bees may encounter during their flight, so that they do not divert from their intended course. This compensatory mechanism is directionally sensitive and is based on the optometric reception of signals. We present a simple schematic diagram in Fig. 1 to explain the course of neural transmission from the photoreceptors to the brain. Correlation between the received signals helps to relate between two distinct points based on location or time, which are essential factors in the locomotion. In this case, we use cross correlation to detect the similarity between the signals from two receptors. Cross correlation between two continuous functions $f_1(t)$ and $f_2(t)$ is mathematically defined as

$$f_1(t)^* f_2(t) = \int_{-\infty}^{+\infty} f_1^*(\tau) \cdot f_2(t + \tau) d\tau. \quad (1)$$

In Fig. 1(b), the temporal filters R_1 and R_2 process the optic signals from the photoreceptors A_1 and A_2 , respectively. A simple delay-and-multiply mechanism works here. The output of filter R_1 , after passing through the temporal filter X_1 , is multiplied (cross correlated) with the signal from adjacent filter R_2 , after it passes through an additional filter Y_2 . Assumption is made that the Y filters are slower in signal filtering than X filters. This leads to a time lag between the incident signals, hence the need for cross correlation arises. A supposed visual pattern moving from left to right will induce signals in receptor A_1 and A_2 with A_1 leading A_2 . The propagation delay in filter Y_2 being more, we get temporal coincidence between signals traversing the path $A_1-R_1-X_1$ and $A_2-R_2-X_2$ resulting in positive response [indicated by + in Fig. 1(b)]. Similar explanation can be given for negative response resulting from a visual pattern moving from right to left. The signals from both the paths, after being correlated, are summed and passed to the elementary motion detector (EMD). The outcome from the EMD is the directionally selective motion response. This relative strength forms the basis of the saccadic flight strategy.

D. Centering Response in Bees

Honeybees can successfully balance the distance between the flanking sides while traversing through a gap. Srinivasan *et al.* [13] studied the centering response through

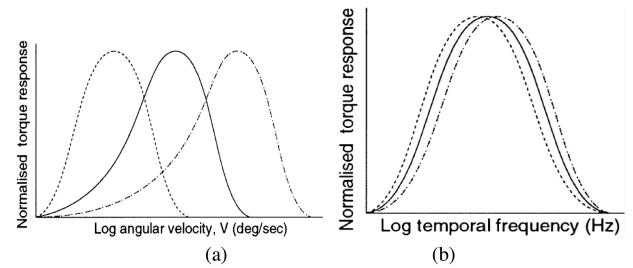


Fig. 2. Observed torque response in the bees versus (a) angular velocity V (degrees $^{-1}$) and (b) temporal frequency F_{temporal} (Hz) as shown by Srinivasan *et al.* [13].

an elaborate setup. It was concluded that the bees avert mid-fight collisions by balancing angular speeds apparently subtended by the walls. Further investigation revealed that the balancing takes place independently of the geometrical pattern (spatial-frequency variation) exhibited by both the sides. The centering response is insensitive to directions while the photoreceptors' response is directionally sensitive in nature and can be implemented in optimization theory to avoid trapping at local optima.

E. Neural Response to Spatial and Temporal Change in Physical Environment in Bees

Srinivasan *et al.* [14] experimentally investigated the optometric response of a bee suspended in a drum and subjected it to rotation of the black and white strip lines in the inner lining of drum. The results have shown that the torque response for the variation of the angular velocity of the drum [as shown in Fig. 2(a)] is a bell-shaped curve. The peak response obtained varies significantly with the change in the angular velocity of the drum, sometimes exhibiting a short tail (low velocity) and at times exhibiting an indented tail (high velocity). This variation in response can be better approximated by using Cauchy distribution (11), which has a long tail compared to the normal distribution. A similar plot of the variation of the torque response is shown in Fig. 2(b) with the temporal frequency results in an identical Gaussian curve (9) but with minimal displacement with the change in the frequency. It must be noted that the variation is with respect to the logarithmic scale. On linearizing them, an exponential variation of the mean is to be included [see (12) and (13)].

F. Saccadic Flight Strategy

Saccade refers to the rapid succession of individual movements of the eye, head, or other body parts that help in extracting the essential information about the spatial structure of environment. It helps in creating a map of the surrounding, and aids in mental analysis. It can be analogously inferred that just like polyatomic molecules most of the flying insects have six degrees of freedom: three translational and three rotational. This complicates the optic flow model and the saccadic flight strategy may help in segregating the translational movement signals from the rotational ones. The pattern of movements displayed by the honeybees help them to obtain a clearer picture of the newly found food sites. They memorize such locations and convey these to the waiting foragers in the hive.

The aforementioned details are fundamental to the locomotion in bees and in turn to the proposed algorithm.

III. CLASSICAL ABC ALGORITHM

The classical ABC algorithm [15], [16] was proposed and investigated for the derivative free optimization of nonlinear, nonconvex, and multimodal objective functions. The steps of the ABC algorithm are outlined as follows.

A. Initialization of Food Source

ABC begins its search for the optimal solution by randomly initializing food sources, θ_i^C representing the i^{th} member, in the real parameter space \mathcal{R}^D . These trial solutions are randomly initialized within the solution space as

$$\theta_{ij}^C = \theta_{j,\min} + \text{rand}_j(0, 1) \times (\theta_{j,\max} - \theta_{j,\min}) \quad (2)$$

where $j = 1, 2, \dots, D$ and $i = 1, 2, \dots, SN$ for a D -dimensional problem employing SN number of sources in the current cycle C . Initial solutions spread across the solution space defined by the bounds $\theta_{\max} = (\theta_{1,\max}, \theta_{2,\max}, \dots, \theta_{D,\max})$ and $\theta_{\min} = (\theta_{1,\min}, \theta_{2,\min}, \dots, \theta_{D,\min})$. The trial counter of each food source is set to 0 after the initialization process. The termination criterion is reached when the value of C reaches the maximum cycle number MCN or the error value attained is less than a predefined threshold. The former is used here.

B. Employer Bee Phase

In this phase, the employed foragers seek a food source with the intention of exploiting it. It is a customary scheme to set only one forager for a given food source such that bijective mapping occurs. The forager exploits the local site and modifies its position around the source by changing a single positional parameter as

$$v_{ij}^C = \theta_{ij}^C + \phi_{ij} \times (\theta_{ij}^C - \theta_{kj}^C) \quad (3)$$

where i is the running index and k can take any values from $\{1, 2, \dots, SN\} \setminus \{i\}$ with ϕ_{ij} being a random number uniformly distributed in the range $(-1, 1)$. ϕ_{ij} determines the degree of positional perturbation of the forager. In the cases of trial solutions overshooting the bounds, they are reinitialized within the bounds. The foragers continually analyze their near vicinity in the search of better food source. They evaluate the fitness based on the nectar content (fitness value). Similarly ABC uses a fitness function formulated (for minimization problems) as

$$\text{fitness}_i = \begin{cases} 1/(1 + f_i) & \text{if } f_i \geq 0 \\ 1 + \text{abs}(f_i) & \text{if } f_i < 0 \end{cases} \quad (4)$$

where f_i is the actual objective (or cost) function to be minimized. For maximization purpose, we can directly set fitness_i equal to f_i . If a food source of higher fitness is found, then greedy selection occurs to replace the present position θ_i^C of its employed food source by the updated position v_i^C . However, in case of no improvement in fitness value, the trial counter is increased by 1 to indicate worsening fitness value.

C. Onlooker Bee Phase

The nectar-collecting foragers after loading nectar travel back to the hive to unload it. In the process, they may communicate the information about the newly found sources to the bees waiting in the hive. Depending on the fitness value conveyed by the waggle dancers, the onlookers select a single food site. This is done by the roulette wheel process where the fitter food sources have more chances of being selected than a less fit one. The probability of getting selected is proportional to the fitness value and is calculated as

$$p_i = \frac{\text{fitness}_i}{\sum_{i=1}^{SN} \text{fitness}_i}. \quad (5)$$

Again, a similar positional modification as in (3) occurs followed by the greedy selection and the trial counter update. This process is continued until all the onlookers have been allotted food sources.

D. Scout Phase: Abandonment Criteria

It is natural for a food source to become exhausted after being repeatedly exploited by the foragers. To avoid making further travels to an exhausted food source, the employed bee abandons it and behaves as a scout bee that performs random walks in the search space to locate a new food source. This analogy is mapped by a trial counter, which is increased by 1 each time the forager is unable to improve upon its previous value. Once it reaches a threshold value defined by the parameter limit , it is thought to be exhausted and the food source is reinitialized randomly and the corresponding trial counter is reset to 0. The parameter limit significantly affects the performance of the ABC algorithm. Small value of parameter limit is identical to premature reinitialization while a considerably larger value leads to wastage of fitness evaluations (FEs) in computing fitness of exhausted food sources with no signs of development.

IV. SIPABC-SF ALGORITHM

This section describes the proposed algorithm in sufficient detail. In Section IV-A, we discuss the locomotion modes that have been implemented in the algorithm. This is followed by a discussion in Section IV-B on the associative-color learning in honeybees that helps to link the locomotion schemes with the color pool. In Section IV-C, we elaborate the saccadic flight strategy and its implementation. Finally, in Section IV-D, our algorithmic model is described along with the pseudocode.

A. Locomotion Schemes: Analysis of Positional Modification

Classical ABC algorithm carries out modification of the forager's position according to (3) whereby a component of a randomly sampled displacement vector is added to the position vector of the presently employed food source. This rectilinear locomotion greatly inhibits the solution domain that is explored. However, it is imperative that the perturbation operation must distribute the search agents over a greater region of the solution space. As depicted in Fig. 3, we see

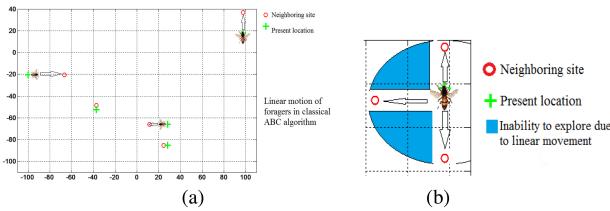


Fig. 3. Search mechanism employed by ABC algorithm. (a) Rectilinear propagation of search agents (foragers) in search space. (b) Narrow exploration range of ABC algorithm search space.

that the foragers move in straight lines. This causes the areas marked in blue, in Fig. 3(b), to remain mostly unexplored. Even to explore this region, say to reach a point in the blue region, it requires a minimum of two consecutive rectilinear perturbations and leads to the wastage of one extra FE. Akay and Karaboga proposed a measure to tackle this in [17]. However, there was little effort to diversify the search process. Diversification does not strictly apply to population diversity only; it can refer to the varying modes by which we perturb the solutions (forager's position) in the solution space as in many EAs [18], [19].

Inspired by this, we use certain locomotion modes to scatter the foragers to ensure improved search. These are but simple modifications on the original ABC scheme. They are incorporated in the algorithmic framework based on the thumb rule that the bees change their position and orientation during foraging; they adopt a variety of motions both rotational and translation. For simplicity, we mimic the translational components through the varying modes of locomotion. The formulas and associated nomenclatures are as follows.

- 1) *Normal spatial locomotion (NRm-SPL):* The normal spatial locomotion is the originally proposed scheme which is same as (3). The forager undergoes a translation directed along the displacement vector from the employed food source θ_i^C to a neighboring one.

$$v_{ij}^C := NRm - SPL(\theta_{ij}^C) = \theta_{ij}^C + \phi_{ij} \times (\theta_{ij}^C - \theta_{kj}^C) \quad (6a)$$

where j is the parameter to be perturbed. Modifying the $NRm - SPL$, we define a set of locomotion modes.

- 2) *Explorative spatial locomotion (ER-SPL):* This mode, as the name suggests, ensures greater exploration of the search space. Only difference is in the random manner in which the mutually exclusive food sources are chosen with respect to the employed food source θ_i^C .

$$v_{ij}^C := ER - SPL(\theta_{ij}^C) = \theta_{k_1 j}^C + \phi_{ij} \times (\theta_{k_2 j}^C - \theta_{k_3 j}^C). \quad (6b)$$

- 3) *Exploitative spatial locomotion (EX-SPL):* The $EX - SPL$ spatial locomotion is a greedy scheme suitable for unimodal functions. It displaces the foragers from its present position and attracts them toward the fittest food source θ_{fit}^C thereby enhancing the rate of convergence

$$v_{ij}^C := EX - SPL(\theta_{ij}^C) = \theta_{ij}^C + \phi_{ij} \times (\theta_{fit}^C - \theta_{ij}^C). \quad (6c)$$

- 4) *Modified approaches:* Using the basic modes, $NRm - SPL$, $ER - SPL$, and $EX - SPL$, we can employ modified locomotion modes. The relative focus on exploration and exploitation is brought about in two ways, either by adding an extra directional component (with an apostrophe ' in the name) during positional perturbation or by decentralizing the attraction from the fittest food source to the top $N\%$. The same nomenclature is applied here with only the case of the letters x and r denoting the relative tendency to exploit or to explore. For instance, with X indicates high affinity for exploitation while x refers to reduced tendency to exploit. Same explanation goes for exploration.

$EXr - SPL(\theta_i^C)$: This locomotion mode is less greedy than the $EX - SPL$ scheme. Instead of using the fittest member, we randomly select one of the fittest $N\%$ sources denoted by θ_{Nfit}^C . It balances between exploitation and exploration with more focus on the former (indicated by X and r)

$$v_{ij}^C := EXr - SPL(\theta_{ij}^C) = \theta_{ij}^C + \phi_{ij} \times (\theta_{Nfitj}^C - \theta_{ij}^C). \quad (7a)$$

$EXR' - SPL(\theta_i^C)$: On adding an extra random displacement vector $(\theta_{k_1 j}^C - \theta_{k_2 j}^C)$ to $EXr - SPL$, we obtain this locomotion mode designed to ensure greater search space exploration while learning from the top $N\%$ food sources simultaneously. The use of symbol X and R (both in upper case) indicate that the strength of exploration is comparatively stronger than the $EXr - SPL$ scheme

$$\begin{aligned} v_{ij}^C := EXR' - SPL(\theta_{ij}^C) &= \theta_{ij}^C + \phi_{ij} \\ &\times \{(\theta_{Nfitj}^C - \theta_{ij}^C) + (\theta_{k_1 j}^C - \theta_{k_2 j}^C)\}. \end{aligned} \quad (7b)$$

$Exr - SPL(\theta_i^C)$: This locomotion mode is an intermediate between $EXr - SPL$ and $ER - SPL$, whereas the employed food source θ_{ij}^C in $Exr - SPL$ is replaced by a randomly selecting one. This helps to distribute attraction more uniformly since a random member experiences attraction to any one of the elite members. This strategy is neither too greedy nor too random and maintains the exploration-exploitation tradeoff

$$v_{ij}^C := Exr - SPL(\theta_{ij}^C) = \theta_{k_1 j}^C + \phi_{ij} \times (\theta_{Nfitj}^C - \theta_{k_2 j}^C). \quad (7c)$$

$ExR' - SPL(\theta_i^C)$: While maintaining the exploitation of $Exr - SPL$ scheme, further diversity can be induced by adding a further directional component to it. It can be presented as

$$\begin{aligned} v_{ij}^C := ExR' - SPL(\theta_{ij}^C) &= \theta_{k_1 j}^C + \phi_{ij} \\ &\times \{(\theta_{Nfitj}^C - \theta_{k_2 j}^C) + (\theta_{k_3 j}^C - \theta_{k_4 j}^C)\}. \end{aligned} \quad (7d)$$

In (6) and (7), k_x are mutually exclusive integers that can take any values from $\{1, 2, \dots, SN\}$ except running index i . We make use of locomotion modes $ER - SPL$, $Exr - SPL$, $Exr - SPL$, $EXR' - SPL$, and $ExR' - SPL$ in our algorithm for modifying the position of the bees.

B. Associative Learning in Honeybees: Color Guided Selection of Locomotion Modes

Frisch [12] and Menzel and Backhaus [11] experimentally established that the bees are exposed to colors on entering

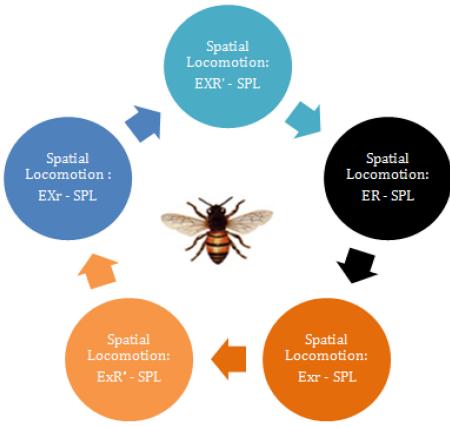


Fig. 4. Array of colors encountered by a forager and the associated motion associated with it.

a floral site and they tend to memorize the color and the associated reward (nectar content). In optimization scenario, an ideal analogy would be to think of the associated reward in terms of fitness of the food source θ_i^C and the color being representative of a locomotion mode that the bee has undertaken to modify its position. Based on this analogy, we make use of five different colors to synergize the locomotion pool with the colored vision of foragers. In Fig. 4, we depict the array of colors encountered by the forager along with the associated locomotion mode. Note that irrespective of the initial exposure to color, bees tend to learn the colors associated with more rewarding food sources gradually through their own experience. To understand this, assume that the foragers are spread in the functional landscape uniformly. Now which particular locomotion mode is helpful in discovering better food sites depends on the locality of the forager, as well as their distribution at that instant. Thus, different foragers learn different colors. So instead of maintaining individual records of each foragers, we use an average metric called color selection mean (C_m), which is updated as shown later.

This raises an important issue: how do the bees perform locomotion based on color learning? A very simple yet effective measure has been put into action by designing a color selection scale (as in Fig. 5) to link the colors with locomotion modes. The scale ranges from 0 to 1 and is divided into five parts, each corresponding to a particular color (same as those in Fig. 4) which in turn is associated with a locomotion mode. The bees usually have a tendency to learn darker color (violet, black) quickly as compared to lighter ones. This behavior has been mapped by allotting a larger portion of the scale for black color. We assume that at the start of the foraging process, the unemployed bees randomly enter the floral sites and encounter one of the five colors and based on it perform spatial locomotion via the associated mode. The value of C_m indicates the mean color response of the entire forager population.

Based on their color learning rate, the gradual transition occurs and the foragers select the locomotion modes probabilistically. We shift the value of C_m by a weighted sum. The color selection probabilities of successful food sources



Fig. 5. Color selection scale indicating proportionate coloring strategy.

are considered during powered mean calculation. The color selection scale closely resembles the natural behavior of bees. Initially, the bees retain their behavior of recognizing dark color, so the initial color selection mean C_m is set 0.5 (shown by the white arrow in Fig. 5). It coincides with the center of the scale corresponding to black representing *ER-SPL* mode. The lighter color variants are present on the right side of the color selection scale and correspond to modes—*EXR'-SPL* and *ExR'-SPL*.

C. Finding Intelligent Directions During Foraging: Saccadic Flight Strategy

A saccade is a rapid intermittent movement of the eye observed in a living organism with the purpose of gaining useful information about the spatial surrounding. Experimental demonstrations [20] suggest that during a flight the bees perform sharp turns of the body and the head to become aware of their present location. Biologically, it has been confirmed that the translation motion is guided by the perceived distance to surrounding obstructions, and the saccadic flight strategy comes handy in optical filtering of the rotational components allowing the translational ones to pass through.

By using local information from similar members we arrange the food sources on the fitness scale, as shown in Fig. 6, to analyze randomly selected food sources used for perturbation of food source θ_i^C . We encode the saccadic flight strategy through two subroutines: 1) tradeoff function $\tau(\cdot)$ and 2) the correlative neurological response $CNR(\cdot)$. $\tau(\cdot)$ is a two-step approach involved with the generation of mutually exclusive food sources selected by the tournament selection and then determining their relative position. It returns a value, which decides the relative position of the selected food sites on the scale, based on which the neurological response is simulated depending on the correlation-based optic model, discussed in Section II-D. Note that we are simulating the photoreceptor model depicted in Fig. 1, but instead of finding the correlation between images we are concerned with the correlation between the food sources, i.e., the trial solutions here. It can be realized in terms of divertive forces in the original locomotion path to guide the foragers along intelligent directions in the search process. The response depends on the value of relative response parameter δ and also on the locomotion modes. The individual subroutines are discussed as follows.

- 1) *Tradeoff function $\tau(\cdot)$:* Movement needs energy to be expended on the part of the foragers to travel to-and-fro its selected site. It is advisory on their part to analyze the distance E and the reward T associated with a given food site and make a tradeoff between them to arrive at a decision of either selecting or discarding the food source. If energy expended in traveling the distance E to a food source is not commensurate the reward T received

from it, the forager will decide against travel. Thus, the need for a tradeoff based on distance and fitness is mandatory. Conventional ABC framework shows that the onlooker bees select a food source based on its relative fitness making it prone to local trapping since multiple onlookers can select a fitter source leading to crowding of foragers around presently found suboptimal solutions. $\tau(\cdot)$ provides an alternative way. It is an operator whose purpose is to generate a set of mutually exclusive food source locations that the onlooker can approach without losing too much energy and being amply rewarded with food. It consists of two subroutines: generator and analyzer. The generator produces an array of exclusive food sources selected by the tournament selection procedure. Similarity is measured in terms of the p -norm distance. Then, the analyzer maps the generated individuals on the fitness scale and assigns them relative position scaled in descending order from the fittest to the least fit one.

- 2) *Correlative neurological response CNR(.)*: Originally correlation model is based on the retinal flow of data from two photoreceptors located in the front end, one being slightly displaced compared to the other. In case of general D -dimensional space, the biological 3-D vision is of no practical use. We apply the relative position of the exclusive food sources, as returned by the tradeoff function, to guide the optic model. The food source located centrally is demarcated as median fit and is set as the reference. Now, the selected mutually exclusive food sources will lie either to the left or right of the reference. Those lying between the best fit and the median are treated as relatively good while those lying between the worst fit and the median are judged to be comparatively bad. An example has been illustrated in Fig. 6 to explain the functioning. Suppose, we make use of an array of 15 food sources. A call to function $\tau(\cdot)$ returns a set of five food sources 5, 8, 9, 13, and 15. In the scale, the food sources are sorted in descending order of fitness value with food source 3 being the fittest one while 12 being the worst one.

The functioning of the feedback mechanism, illustrated in Fig. 1(b), is now discussed to generate corrective motion component (Section IV-D) using the T/E ratio to guide the bee intelligently in the search space. Let N_n and N_p denote the number of food sources lying to the right and left of the median fit position (7th food source in Fig. 6), respectively. The value of relative response parameter ($\delta = N_p - N_n$) gives the difference between food sites that are treated as fitter N_p to those that are relatively less fit N_n . Strong positive values of δ (2 or 3) gives positive feedback thereby simulating the + signal [in Fig. 1(b)], while a sufficiently negative value (-2 or less) evokes the - signal. A value lying in the range of [-1, 1] does not give a strong response. Table I maps the locomotion mode with the corresponding neural response based on the value of δ .

The correlative response results in generation of corrective motion components that are added to the original locomotion

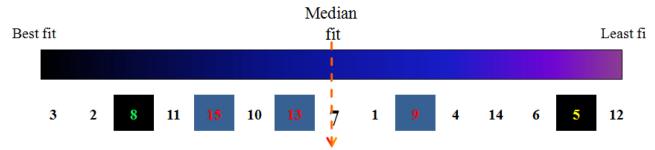


Fig. 6. Relative position of the food sources on fitness scale. It is affirmed that the position of food source 8 (marked in green) is nearest to the best fit side is the fittest one selected while 5 (marked in red) being nearest to the least fit side of the scale is judged as the least fit. The rest of the selected food sources are marked in red denoting intermediate fitness.

TABLE I
CORRELATIVE NEUROLOGICAL RESPONSE BASED ON LOCOMOTION
MODE AND VALUE OF δ

Mode	δ	Correlation	Response
$EXr-SPL$	$\delta = 0$	Moderate	Centering
$EXR'-SPL$	$\delta < 0$	Negative	Escaping
	$\delta > 0$	Positive	Seeking
$ER-SPL$	$\delta \geq 1$	Positive	Seeking
	$\delta \leq 0$	Negative	Escaping
$Exr-SPL$	$\delta \geq 1$	Moderate	Seeking
$ExR'-SPL$	$\delta \leq 0$	Negative	Centering

to enhance diversity. How and why they are implemented is discussed in the next subsection.

An important observation was made with respect to this T/E tradeoff since it involves both fitness and distance calculations. The authors state that by replacing the fitness-based selection of food source in classical ABC approach by making use of distance E affects the algorithmic performance. This prompted us to simulate the T/E tradeoff by using affinity-based (distance) probability calculation (16). After selecting the food sources, they are mapped in fitness scale to determine N_n and N_p and in turn δ . The corrective motion generated by CNR makes use of the T/E ratio by adding directional components directed along $\theta_{bestfit}$ or $\theta_{worstfit}$, which are food sources that have the highest and lowest value of T/E ratio with respect to θ_i , respectively. This small change was instrumental in enhancing not only the convergence speed but also helped to reduce the chances of premature trapping for deceptive functions where most of direct search methods fail. Having discussed the essential components of our method, we proceed with the description of the algorithm.

D. SiPABC_Sf Algorithm

In our proposed SiPABC_Sf approach, the similarity with the classical ABC algorithm is kept intact with vital modifications in the operator dynamics. Unlike classical ABC, we do not distinguish between the employed bee and the onlooker bee phase. Rather we operate on the same forager population in two stages: 1) the color guided locomotion (CGL) and 2) the saccadic flight strategy (SFS). We felt that instead of evaluating the trial population twice (after two phases) as is done in ABC, it is judicious to use the CGL to modify the forager's position and then a corrective motion component

based on the locomotion mode chosen earlier. Only then fitness evaluation occurs and synchronous greedy selection is instantiated. This has benefits as far as population diversity and time complexity issues are concerned. Also a modified affinity-based probability calculation has been applied.

Just like any population-based metaheuristic, the first step begins with the instantiation of a trial population using (1). The default colony size SN is set to 100 for our algorithm. This initialized trial population will be improved upon through the repeated application of search operators iteratively until the termination criterion is met. The details of the individual stages, as used in our algorithm, are given as follows.

1) Color Guided selection of Locomotion Pool (CGL): The color selection probability CS_i is generated independently from normal distribution for every forager as

$$CS_i = \text{Gaussian}(Cm, \sigma_m) \quad (9)$$

where the initial mean of the Gaussian distribution is equal to 0.5 and the standard deviation σ_m equal to 0.2. The value of CS_i is truncated in the range [0, 1] and is regenerated otherwise. The value of CS_i is compared with the color selection scale (shown in Fig. 5) and accordingly the locomotion mode is chosen as presented as follows:

$$\begin{array}{ccccc} (0, 0.1] & (0.1, 0.3] & (0.3, 0.7] & (0.7, 0.9] & (0.9, 1.0] \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ EXr-SPL & Exr-SPL & ER-SPL & ExR'-SPL & EXR'-SPL \end{array} \quad (10)$$

While performing locomotion, the degree of positional modification of the j th parameter of i^{th} food source is controlled by the perturbation factor $\phi_{i,j}$ sampled from Cauchy distribution

$$\phi_{i,j} = \text{Cauchy}(\mu_p, \sigma_p) \quad (11)$$

where μ_p and σ_p are the location parameter and scale parameter of Cauchy distribution having initial values of 0 and 0.2. The value of scale parameter is varied nonlinearly as

$$\sigma_p(C) = \sigma_p(0) - (\Delta\sigma) \left(\frac{e^{m.C/MCN} - 1}{e^m - 1} \right) \quad (12)$$

with the value of $\sigma_p(0)$ (initial), $\Delta\sigma$ and m being 0.2, 0.1 and 10, respectively. The scale parameter variation is based on the response to angular velocity [as in Fig. 2(a)], which is nonlinear in nature. Due to infinite second moment the Cauchy distribution has a wider tail than normalized distribution and is used for approximating the torque response. We point out that the nonlinear variation, as is (12), accounts for the logarithmic nature of the x -axis as is observed in Fig. 2(a). The value of MR is kept fixed at 0.65 in our approach. The value of the mean Cm is updated after each cycle as

$$Cm = w.Cm + (1 - w).mean_{pow}(CS^*) \quad (13a)$$

where the weighting factor w is randomly generated in the range (0.95, 1] and $mean_{pow}$ is the powered mean of the set CS^* of CS_i of successful solutions. The power p is varied as per the temporal frequency response [Fig. 2(b)]

$$p(C) = p(0) + (\Delta p) \left(\frac{e^{m.C/2.MCN} - 1}{e^{m/2} - 1} \right). \quad (13b)$$

Equation (13b) is identical to (12) with the initial value $p(0)$ being 1 which is increased to a maximum of 2 (Δp being 1). The mean Cm is reflective of the performance of the locomotion modes at play and gives a truer indication of the population. This nonlinear variation [21] is essential to the performance of the algorithm. The value of N used in (6) is varied similarly from 35% to 10%.

2) Saccadic Flight Strategy: Following the CGL, we calculate the affinity matrices D and F which holds the distance and fitness information relative to each forager. Note that this calculation can be done by using only the upper triangular part of the matrices based on the property of symmetry and skew symmetry. That is to say the elements of matrices D and F satisfy the condition $D(i, j) = D(j, i)$ and $F(i, j) = -F(j, i)$, $j, i \in \{1, 2, \dots, SN\}$, respectively. We also need to keep a copy of the population members in descending order of fitness value. Even the order of indices can also be stored saving memory space. This will be utilized by the tradeoff function later on.

With reference to i^{th} food source, we will proceed in our discussion. During the matrix computations, we determine the probability of selecting the j th food source for the positional modification of θ_i^C . It is calculated as

$$p_{i,j} = 0.9 \frac{D(i, j)}{\max_j \{D(i, j)\}} + 0.1 \quad (14)$$

where $D(i, j)$ refers to the element of the j th column in the i th row. Depending on the locomotion mode that was applied to θ_i^C in the previous stage, x food sources are selected with the probability value calculated above, i.e., $\text{rand}[0, 1] \leq p_{i,j}$. The variable x equals the number of stochastic food sources associated with a locomotion mode as given in (6b) and (7). So x takes the value of 0 for $EXr-SPL$, 2 for $EXR'-SPL$ and $Exr-SPL$, 3 for $ER-SPL$ and 4 for $ExR'-SPL$. This is done by the generator part of the tradeoff function.

The selected x food sources are mapped on the fitness scale (*if $x > 0$*). The mapping is enacted through a simple logical process. The indices are stored in a temporary archive based on fitness value and first, last, and the median fitness values are retrieved from it. A one to one check occurs in which the fitness values of each of the x food sources are checked. If it lies between the fitness value of median and first element, then variable N_p is incremented by 1. On the other hand if it is below the fitness of median element, then N_n is incremented by 1 such that $N_p + N_n = x$. The value of $\delta = N_p - N_n$ is calculated and based on the entries given in Table I, the corrective motion response is chosen. However, the motion component is the result of the correlation-based optic model that in response to the local neighborhood. The response formulas are given as follows:

$$\begin{aligned} \text{Seeking Response} & \quad dir_s = \alpha_+ (\theta_{bestfit} - \theta_i) \\ \text{Escaping Response} & \quad dir_e = \alpha_- (\theta_i - \theta_{worstfit}) \\ \text{Centering Response} & \quad dir_c = \alpha_+^1 (\theta_{bestfit} - \theta_i) + \alpha_-^1 (\theta_i - \theta_{worstfit}) \end{aligned} \quad (15)$$

where $\theta_{bestfit}$ and $\theta_{worstfit}$ refer to the fittest and the worst food sources based on the T/E ratio. For centering response, information from the best and the worst food sources is

Algorithm 1 Pseudo-code of SiPABC_Sf algorithm

```

begin
1:   for i=1 to SN
2:     Create food source  $\theta_i^C$  based using Eq. (1). //Initialization
3:     Evaluate its fitness using Eq. (3). // Fitness evaluation
4:   endfor
5:   Set the variables: countFES = SN, max_iter = maxFES / SN, iter = 1, m = 1, Cm = 0.5, N = 0.35 * SN
6:   while countFES < max_iter //iteration
//Color guided selection of locomotion mode.
7:     for i = 1 to SN
8:       Generate color selection probability  $CS_i$  using Eq. (7).
9:       Depending on the value of  $CS_i$ , obtain locomotion mode using Eq. (8) and save it.
10:      Modify the position of the  $i^{th}$  food source as  $v_{ij}^C := Mode(\theta_{ij}^C) * \text{with probability } MR$  (as per Eq. 6).
11:    endfor
// Matrix calculations.
12:    Compute the matrices D and F using the properties of symmetry and skew-symmetry.
13:    Calculate the probability values using information from matrix D.
14:    Obtain the list of indices sorted in descending order of fitness value.
// Saccadic flight strategy
15:    for i = 1 to SN
16:      Depending on the  $Mode(\theta_{ij}^C)$ , saved earlier, select x food sources as described in Section III-D (part ii).
17:      Set value of relative response parameter ( $\delta$ ) to 0.
18:      if  $x > 0$  Compare the obtained food sources using the sorted indices list (obtained in 14:) and update  $\delta$ .
19:      Refer to Table I and based on  $Mode(\theta_{ij}^C)$  and  $\delta$ , obtain the response that needs to be implemented.
20:      Update the position of the food source  $\theta_i^C$  by modifying the parameters that remained unchanged  $\dagger$  with probability  $1 - MR$  as
        for j = 1 to D
          if rand <  $1 - MR$  &&  $v_{ij}^C == \theta_{ij}^C$ 
             $v_{ij}^C := Mode(\theta_{ij}^C) + Response^\odot$ 
          endif
        endfor
21:      Evaluate the position of the updated food source and store it in a temporal archive.
22:    endfor
23:    Perform population update through synchronous greedy selection.
24:    Update the parameters using Eq. (12) to (14).
25:  endwhile
end

```

* $Mode(\cdot)$ can be any one of the locomotion modes defined in Eq. (6b) and (7).

\dagger The condition for checking has been given in if-statement. This is done to direct corrective motion only along selective dimensions.

\odot This is same as the response outlined in Eq. (15).

essential for directional insensitivity. The scaling factors have been set as

$$\alpha_-^1 = |f(\theta_{bestfit}) - f(\theta_i)| / |f(\theta_{bestfit}) - f(\theta_{worstfit})|$$

$$\alpha_-^2 = -|f(\theta_{worstfit}) - f(\theta_i)| / |f(\theta_{best}) - f(\theta_{worst})|$$

and α_+ are α_- computed as per (11).

Note that due to the saccadic flight strategy, bees do not travel in rectilinear paths and the actual distance traversed by the bees while traveling between food sources is more than the displacement between them. Thus, we use a p -norm distance with $p=1.5$ instead of $p=2$ (Euclidean) for calculating $TE_{i,j} = (f(\theta_j) - f(\theta_i)) / \| \theta_j, \theta_i \|_{p=1.5}$. This gives a near estimate of the distance traveled by the foragers. Relaxation of choice permits uniform distribution of the work-force and saves additional computation of the scout phase used in the original ABC. The generalized formula for the movement of the bee in the onlooker phase is as follows:

$$v_{ij}^C := Mode(\theta_{ij}^C) + Response(r) \quad (16)$$

where $Mode(\theta_{ij}^C)$ is based on color learning as previously mentioned and $Response(r)$ can be any of those given in (15). However, uncontrolled use of this corrective motion component causes the search to be too random in nature. To

check its usage, we apply (16) with a probability of $1 - MR$ on those parameters which were not affected during the positional modification in CGL earlier on. The saccadic flight strategy (SFS) in insect colony is the key to their efficient extraction of neighborhood information and helps in enhancing the search process.

3) *Synchronous Greedy Selection*: This brings us to the last phase of our algorithm whereby synchronous greedy selection is performed. This has its advantages when compared with asynchronous selection. The synchronous technique uses a memory archive to store the solutions generated after the two previous stages have been implemented. The fitness evaluation occurs only after modifying all the food sources. A one-to-one selection occurs whereby the foragers replace the location of the food source in their memory if a better solution is produced by the modified food source. In addition to diversity, the synchronous selection relaxes the need for recalculating the matrices D and F , as well as sorting the fitness matrix every time a solution replaces its predecessor. This will be highlighted later through complexity analysis.

4) *Putting It All Together*: A complete pseudo-code of the algorithm is given below. This will help in getting a complete

TABLE II

MEAN, STANDARD DEVIATION, AND P -VALUES OF WILCOXON'S RANK SUM TEST FOR BEST OF RUN ERROR VALUES FOR 50D FUNCTIONS

Algo	SaDE	JADE	jDE	HPA	ABC	CLPSO	GSO	m-ABC	SiPABC_Sf
Func	Mean (Std) [p -value]								
f_1	1.4872e-42 (2.4190e-40) [3.21e-08]	7.4651e-43 (4.9946e-39) [1.05e-08]	3.1544e-37 (8.2312e-25) [2.79e-13]	3.5455e-21 (3.1299e-20) [3.78e-18]	3.9586e-26 (5.4419e-27) [2.7457e-14]	9.6633e-14 (2.73e-24)	1.6580e-06 (7.3408e-08) [1.91e-34]	6.5369e-37 (4.0194e-42) [1.14e-13]	5.6843e-47 NA
f_2	4.3603e-03 (1.6830e-03) [8.26e-09]	1.6310e-04 (7.8233e-03) [1.25e-08]	1.5902e-02 (1.1286e-03) [1.83e-12]	4.2189e-02 (3.2356e+02) [7.22e-16]	2.9570e+04 (1.2134e+04) [2.37e-18]	3.1925e+03 (7.1422e+02) [7.26e-18]	6.0926e+04 (2.3285e+04) [1.14e-18]	5.2400e-04 (2.2231e-05) [1.15e-07]	1.9744e-07 7.1562e-09 NA
f_3	1.0650e+06 (7.0330e+04) [6.38e-09]	5.0156e+04 (9.6847e+03) [5.81e-03]	3.1070e+07 (4.8640e+06) [3.82e-07]	1.4790e+07 (1.2318e+07) [9.23e-08]	1.5518e+08 (2.7136e+07) [4.82e-09]	1.3272e+07 (5.0194e+06) [9.72e-07]	7.7343e+07 (3.5464e+07) [8.71e-08]	5.1260e+06 (1.5272e+05) [1.09e-05]	1.6919e+04 1.3295e+04 NA
f_4	1.0567e+02 (7.5371e+01) [2.73e-17]	1.0162e+00 (5.5515e-01) [1.98e-10]	1.0194e+04 (9.1828e+02) [3.29e-18]	3.8792e+03 (2.1602e+03) [6.46e-16]	4.1402e+04 (4.1410e+03) [4.99e-18]	7.9909e+03 (1.9105e+03) [3.87e-18]	9.2658e+04 (7.1483e+03) [5.67e-20]	4.1763e+02 (6.4418e+01) [1.01e-14]	2.6636e-03 NA
f_5	2.9988e+03 (7.2840e+02) [2.55e-04]	1.3250e+03 (3.7950e+02) [1.15e-03]	5.0083e+03 (3.4196e+03) [1.84e-07]	6.8296e+03 (3.6752e+03) [9.33e-04]	3.4334e+03 (5.8657e+02) [3.74e-05]	5.7648e+03 (8.0773e+02) [6.29e-04]	2.3662e+04 (2.2227e+03) [6.51e-04]	1.4085e+04 (9.2603e+02) [8.77e-04]	1.0638e+03 4.2642e+02 NA
f_6	1.2433e+01 (7.4640e+00) [7.25e-17]	7.0413e+00 (2.0042e+00) [1.15e-16]	4.2758e+01 (1.0169e+00) [1.27e-17]	9.6714e+01 (4.4244e-01) [7.22e-08]	2.5675e+06 (4.5336e+04) [8.38e-24]	1.0744e+02 (5.9242e+01) [1.70e-21]	8.4898e+08 (1.2879e+09) [1.85e-34]	3.7675e+03 (8.2684e+02) [2.61e-22]	2.7430e-05 NA
f_7	6.1948e+03 (7.5940e-09) [6.82e-20]	6.2532e+03 (6.7486e+00) [5.45e-20]	6.3414e+03 (2.4960e+01) [3.61e-20]	1.4567e-01 (7.5464e+01) [5.12e-05]	8.2911e+03 (1.1576e+02) [5.72e-18]	6.3181e+03 (2.0895e+01) [3.68e-18]	6.2055e+03 (3.3784e+00) [4.57e-20]	3.1953e-01 (3.2047e+04) [1.44e-06]	6.0923e-04 1.2507e-05 NA
f_8	2.1132e+01 (3.4580e+01) [1.16e-02]	2.1131e+01 (2.6510e+02) [1.51e-02]	2.1289e+01 (3.89e-02)	2.1142e+01 (2.5157e+02) [1.09e-02]	2.1165e+01 (1.0069e+00) [8.56e-03]	2.1276e+01 (4.0057e+02) [5.22e-03]	2.1158e+01 (1.5652e+02) [9.91e-03]	2.1120e+01 3.8873e-02 NA	4.5475e-14 NA
f_9	1.1480e+02 (1.2660e+01) [8.92e-25]	1.3520e+02 (2.5910e+01) [1.23e-26]	1.7160e+02 (1.4090e+01) [2.78e-07]	8.3342e-02 (5.3422e-02) [1.89e-09]	2.6679e+02 (3.2175e+01) [3.77e-29]	1.8406e+01 (4.1754e+00) [1.89e-22]	3.0171e+02 (2.7324e+01) [1.22e-31]	6.1887e-08 (8.7140e-08) [2.67e-06]	4.5475e-14 1.2511e-14 NA
f_{10}	6.3420e+01 (1.2870e+01) NA	1.9350e+02 (2.0600e+01) [2.64e-05]	3.9597e+02 (5.6230e+01) [2.78e-07]	2.0934e+02 (1.3421e+01) [4.66e-07]	4.4697e+02 (3.0193e+01) [1.89e-06]	1.8699e+02 (4.1754e+00) [3.71e-04]	6.2579e+02 (3.7428e+01) [1.56e-11]	4.7491e+02 (1.4198e+02) [5.41e-08]	1.3417e+02 (2.1883e+01) [9.15e-03]
f_{11}	6.6340e+01 (1.485e+00) [5.83e-05]	6.2080e+01 (1.7440e+00) [4.38e-04]	7.3300e+01 (1.0080e+00) [2.63e-04]	7.0268e+01 (3.1023e+00) [8.54e-07]	7.4269e+01 (1.5541e+00) [3.74e-05]	6.2320e+01 (2.1217e+00) [2.78e-06]	6.8566e+01 (3.0434e+00) [1.85e-05]	7.2352e+01 (1.3786e+00) [3.61e-06]	4.2031e+01 (1.2312e+00) NA
f_{12}	8.6980e+03 (8.4520e+03) [1.76e-05]	1.6980e+05 (5.1980e+04) [1.05e-15]	1.5136e+05 (1.9580e+05) [2.52e-15]	5.6323e+04 (4.6233e+04) [344e-12]	3.7994e+05 (9.1289e+04) [3.82e-17]	4.9547e+04 (3.1270e+04) [8.46e-12]	7.6201e+05 (2.5427e+04) [1.62e-18]	8.8935e+04 (4.0709e+03) [4.55e-14]	1.8118e+03 (1.1111e+03) NA
f_{13}	1.7710e+01 (6.1120e+00) [2.89e-04]	2.7612e+00 (9.0840e-01) [8.91e-04]	2.9603e+01 (1.3220e+00) [4.4539e+00]	5.3224e+00 (4.4539e+00) [3.22e-05]	5.7648e+02 (1.6313e+02) [7.46e-06]	3.8044e+02 (5.8831e-01) [2.88e-05]	4.0614e+01 (1.2514e+01) [2.61e-05]	7.0973e+01 (3.6495e+00) [1.02e-05]	2.6658e+00 (1.0286e-01) NA
f_{14}	2.1997e+01 (2.0634e-01) [4.72e-02]	2.3146e+01 (2.5685e-01) [8.61e-03]	2.3119e+01 (2.8457e-01) [2.08e-02]	2.2845e+01 (5.3245e-01) [2.87e-02]	2.2629e+01 (1.2810e-01) [3.85e-02]	2.2652e+01 (2.8921e-01) [2.84e-02]	2.2102e+01 (8.2407e-01) [5.16e-02]	2.2994e+01 (1.1620e-01) [1.14e-02]	2.1469e+01 (5.2103e-02) NA
f_{15}	3.9827e+02 (1.0755e+02) [2.83e-03]	2.7690e+02 (8.9640e-01) [8.74e-03]	4.0000e+02 (0.0000e+00) [2.75e-03]	2.3489e+02 (3.8732e+01) [8.64e-06]	4.8795e+02 (9.1882e+00) [6.31e-03]	4.0000e+02 (5.9918e-14) [2.89e-03]	8.0007e+02 (1.3666e+02) [1.56e-08]	6.3689e+02 (2.6355e+01) [5.41e-06]	1.3950e+02 (1.2586e-16) NA
f_{16}	1.4872e+02 (2.8335e-02) [3.21e-08]	1.4370e+02 (5.4257e+01) [4.23e-08]	3.1544e-37 (4.9946e-39) [2.79e-27]	3.7543e+02 (4.2398e+01) [4.35e-10]	9.3872e+02 (5.4419e-27) [3.78e-18]	2.7457e+02 (7.2457e-27) [2.73e-24]	4.0002e+02 (1.1806e+02) [2.94e-16]	4.0002e+02 (1.8419e-02) [2.81e-15]	5.0628e+01 (1.2563e+01) NA
f_{17}	4.3603e+03 (1.6830e-06) [8.26e-19]	1.9023e+02 (3.8743e+01) [1.83e-19]	1.5902e+02 (1.1286e-03) [1.83e-12]	3.3564e+02 (9.475e-03) [2.37e-17]	2.9570e+04 (1.2134e+04) [2.73e-18]	3.1925e+03 (7.1422e+02) [2.76e-18]	6.3521e+02 (7.7593e+01) [1.51e-17]	3.0867e+02 (1.6185e+01) [1.62e-15]	1.0940e+02 (4.0034e+01) NA
f_{18}	1.0650e+06 (7.0330e+04) [6.38e-18]	9.2060e+02 (1.8930e+00) [1.35e-05]	3.1070e+07 (4.8640e+06) [1.02e-03]	9.4937e+02 (2.7136e+07) [4.02e-03]	1.5518e+08 (5.0194e+06) [4.82e-24]	1.3272e+07 (3.7003e+01) [3.72e-20]	1.2158e+03 (4.51e-04) [4.51e-04]	9.9172e+02 (3.7003e+01) [9.81e-04]	8.3984e+02 (6.2592e+00) NA
f_{19}	1.0567e+02 (7.5371e+01) [2.73e-17]	9.2613e+02 (6.51e-06) [1.84e-07]	1.0194e+04 (3.29e-18) [4.75e-05]	9.5028e+04 (4.2034e+01) [5.32e-04]	4.1402e+04 (4.1410e+03) [5.32e-04]	4.7990e+03 (1.9105e+03) [3.74e-05]	1.2279e+03 (2.6464e+01) [3.61e-09]	9.7172e+02 (1.2921e+00) [1.20e-06]	8.3467e+02 (4.2495e+00) NA
f_{20}	2.9988e+03 (7.2840e+02) [2.55e-04]	9.2672e+02 (1.0675e+00) [6.51e-03]	5.00083e+03 (3.4196e+03) [1.84e-07]	9.3874e+02 (1.2343e+01) [5.32e-04]	3.4334e+04 (5.8657e+02) [3.74e-05]	5.7648e+03 (8.0773e+02) [6.29e-04]	1.2539e+03 (3.0142e+01) [1.03e-04]	9.6772e+02 (2.5919e+01) [1.52e-03]	8.3906e+02 (1.0092e+00) NA
f_{21}	1.2433e+03 (7.4640e+00) [7.25e-17]	8.5230e+02 (3.5175e+02) [5.62e-09]	4.2758e+03 (1.0169e+00) [1.27e-17]	5.0000e+06 (3.4321e-06) [8.38e-24]	2.5675e+06 (4.5336e+04) [8.38e-24]	1.0774e+03 (5.9242e+01) [1.70e-21]	1.3431e+03 (3.5842e+01) [2.64e-11]	8.5600e+02 (2.1772e+02) [3.44e-09]	5.0000e+02 (0.0000e+00) NA
f_{22}	9.7245e+02 (3.3383e+01) [2.71e-05]	9.3890e+02 (1.4235e+01) [1.65e-04]	9.8920e+02 (2.0851e+01) [5.84e-06]	1.0964e+03 (4.3278e+01) [4.82e-07]	9.8339e+02 (8.9899e+00) [5.3601e+00]	1.3941e+03 (2.5470e+01) [2.64e-08]	1.5623e+03 (1.5983e+02) [3.69e-09]	1.2064e+02 (3.1072e+01) [NA]	8.2064e+02 (3.1072e+01) NA
f_{23}	6.4230e+02 (4.0890e+01) [1.31e-05]	8.1030e+02 (2.4572e+02) [2.66e-06]	6.3044e+02 (2.9787e+01) [1.63e-05]	5.4012e+02 (3.1289e+00) [8.58e-05]	9.8514e+02 (5.9692e+01) [2.79e-08]	7.3972e+02 (4.2398e+01) [8.27e-07]	1.3515e+03 (3.4894e+00) [2.64e-12]	6.2542e+02 (5.5923e+01) [7.66e-04]	5.0000e+02 (0.0000e+00) NA
f_{24}	2.0000e+02 (0.0000e+00) NA	2.0000e+02 (0.0000e+00) NA	1.0178e+03 (1.8934e+02) NA	8.6523e+02 (3.4589e+01) NA	1.0950e+03 (4.2238e+02) NA	9.8529e+02 (6.1248e+01) NA	1.2350e+03 (2.9283e+02) NA	2.0000e+02 (0.0000e+00) NA	2.0000e+02 (0.0000e+00) NA
f_{25}	1.7586e+03 (3.1453e+00) [2.75e-08]	1.6632e+03 (5.5842e+00) [2.87e-08]	1.7280e+03 (6.2562e+00) [1.63e-06]	1.0856e+03 (8.7539e+02) [7.84e-05]	1.2187e+03 (1.8899e+01) [2.98e-06]	1.6821e+03 (2.4809e+02) [2.81e-08]	9.0153e+02 (4.9213e+01) [2.36e-05]	2.6003e+02 (5.5895e+01) [9.98e-03]	2.0000e+02 (0.0000e+00) NA

picture when all the individual processes are assembled together as one.

V. EXPERIMENTAL SETUP AND SIMULATION RESULTS

A. Benchmark Test Suite

The benchmark suite of IEEE CEC 2005 competition comprises of 25 challenging problems that presents, in addition to scalability, shifting, rotation, and ill-conditioning, highly multimodal landscapes, a huge number of local optima, and nonseparable nature of functions. The first 5 functions $f_1 - f_5$ are unimodal, $f_6 - f_{12}$ are classic multimodal, $f_{13} - f_{14}$ are expanded functions and functions $f_{15} - f_{25}$ are hybrid

composition functions that are formed by combining classical benchmark functions, such as Rastrigin, Schwefel, Griewanks, Weierstrass, etc. The functions also have global optima on bounds, wide optima basin, narrow optima basin, and rotated with high condition number matrix. For more details on these benchmarks, readers may consult the technical report [22]. All problems are scalable and we have considered 30-D, 50-D, and 100-D functions for testing the proposed method.

B. Contending Algorithms Compared and Parametric Setup

The algorithms considered in the comparative study are based on a wide variety of features, such as self adaptation, local search, memory archive, diversification, learning

TABLE III
ADJUSTED P -VALUES FOR PAIR WISE STATISTICAL COMPARISONS BETWEEN SiPABC-SF AND PEER ALGORITHMS

D	Procedure	SaDE	JADE	jDE	HPA	ABC	CLPSO	GSO	m-ABC
30	Holm	0.0037	0.0047	0.0025	6.4e-4	5.8e-18	8.7e-9	3.4e-21	6.8e-9
	Schaffer	0.0041	0.0049	0.001	4.9e-4	1.4e-19	4.9e-9	1.9e-22	8.7e-9
	Nemeny	0.0052	0.0061	0.0039	5.2e-4	5.4e-18	5.6e-10	9.4e-21	4.9e-10
	Bergman-Hommel	0.0049	0.0073	0.0021	5.9e-4	4.6e-18	3.6e-9	8.7e-21	2.9e-10
50	Holm	1.9e-4	2.3e-4	2.7e-6	7.2e-7	2.9e-20	2.8e-10	1.6e-25	1.9e-10
	Schaffer	2.2e-5	3.1e-4	3.1e-6	5.1e-8	1.8e-21	3.4e-11	3.4e-25	3.9e-10
	Nemeny	1.7e-5	1.9e-5	1.2e-6	6.5e-7	5.4e-20	1.9e-10	3.8e-25	5.5e-11
	Bergman-Hommel	1.8e-6	4.1e-4	1.1e-6	8.1e-7	6.7e-20	6.4e-11	1.9e-26	8.1e-10
100	Holm	1.5e-6	5.6e-5	1.1e-7	5.3e-10	2.8e-24	3.6e-13	3.4e-28	6.4e-14
	Schaffer	2.1e-6	7.2e-5	1.2e-7	4.9e-11	1.9e-23	7.8e-13	5.6e-28	2.8e-14
	Nemeny	3.7e-7	4.9e-5	2.2e-8	7.5e-10	1.8e-24	2.5e-14	4.5e-29	3.7e-13
	Bergman-Hommel	1.9e-7	1.1e-6	3.5e-8	5.9e-11	1.1e-24	1.9e-14	1.9e-30	9.1e-13

strategy. Each algorithm has its unique property of guiding the search toward promising regions of the fitness landscape. Best parametric settings were used for the peer algorithms as recommended by the corresponding literatures.

- 1) Artificial Bee Colony (ABC) [5], [16].
- 2) Modified Artificial Bee Colony (m -ABC) [10].
- 3) Differential Evolution with Strategy adaptation (SaDE) [23].
- 4) An adaptive DE with optional external archive (JADE) [24].
- 5) A Self-adaptive DE (jDE) [25].
- 6) Hybrid approach based on PSO and ABC algorithm (HPA) [26].
- 7) Comprehensive Learning PSO (CLPSO) [27].
- 8) Group Search Optimizer (GSO) [28].

For analysis, we have conducted 50 independent sample runs of the contending algorithms for each function and reported the mean and the standard deviation of the best-of-run error values. The simulation was carried out under the following platform configurations.

- Software: MATLAB 2010a (version 7.10).
- Processor: Intel i5 2nd gen.
- CPU Speed: 2.76 GHz.
- Memory: 4-GB DDR3 RAM.
- Operating System: MS Windows 7.

The best-of-run error has been defined as the difference between the actual global optima f^* of the objective function under consideration and the best functional value $f(\theta_{\text{best}})$ achieved by the optimizer after a given FEs budget, which marks the termination criterion. The termination criterion can be any one of the following-time budget (online optimization), fixed number of iterations (offline optimization) or fixed number of FEs (engineering problems). According to the technical report [22], the stopping condition is maximum number of FEs which is fixed at $D \times 10^4$ for D -dimensional problem. To understand the statistical significance of the results, we use Wilcoxon's rank-sum test [29] on independent samples at 5% significance level. The P -values returned by the test must be less than 0.05 (5% significance level) as a proof against the null hypothesis. Additional inclusion of Nemeny's test, Shaffer's static procedure, Holm's test and Bergmann-Hommel's dynamic procedure help to perform multiple pair

wise comparisons. The adjusted P -values are obtained through these four procedures as reported in [30] and [31].

C. Comparative Analysis

Experimental observations in terms of the mean and standard deviation of the best-of-the-run errors are recorded. Due to space limitation, we have presented the result for 50 dimensions in Table II for each of the contending algorithms on functions f_1 to f_{25} . Discussions on 30-D and 100-D have been provided in the supplementary document. P -values obtained through the rank-sum test between the best algorithm and each of the remaining algorithms over all the benchmark functions are also presented in these tables. In these tables, NA stands for not applicable and occurs for the best performing algorithm itself in each case. Besides, we have included the result of the aforementioned four statistical tests in Table III. 30-D, 50-D and 100-D have been considered to present a general overview.

Table II indicates that the performance of SiPABC-Sf is remains overall commendable in 50-D search space. In fact, it excels in 50-D domain by achieving lower mean errors compared to its contenders on 23 instances. On f_{10} SaDE manages to outperform SiPABC-Sf which attains second ranks among the participating EAs. SiPABC-Sf presents a dominating performance for the unimodal (f_1 to f_5) and classic multimodal problems (f_6 to f_{12}). For the expanded functions, it performs at par with other algorithms and achieves error of the same order. As the dimensionality is scaled up, solving the hybrid composition functions (f_{16} to f_{25}) become a major challenge due to the exponential rise in local peaks and high dispersion metric of the landscape. Most of the optimizers are trapped in deceptive regions resulting in high error values. SiPABC-Sf puts up a satisfying performance and obtains least mean errors in a majority of cases. Only for rotated hybrid composition f_{24} , the results obtained are at par with adaptive DE variants-SaDE, JADE and jDE, CLPSO and m-ABC, with all reaching the value of 2e+2.

The incorporation of a learning mechanisms (in CLPSO), strategy or parameter adaptation (in algorithms like SaDE, JADE, and jDE) and even hybridization (HPA) helps to avert the problem of dimensionality to a good extent. Algorithms, focusing only on local search, like ABC fail to locate optimum when the dimension increases. CLPSO's performance might

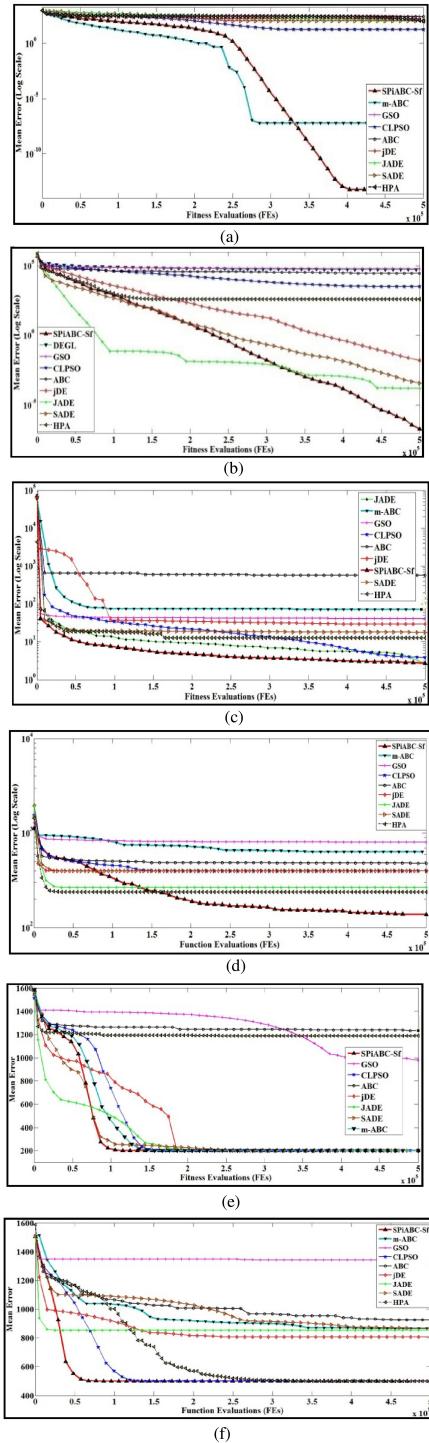


Fig. 7. Median error of competing algorithms plotted against increasing FEs for six 50D benchmark problems. (a) f_2 : Shifted Schwefel's function. (b) f_9 : Shifted Rastrigin's function. (c) f_{13} : Expanded function. (d) f_{15} : Hybrid composition function. (e) f_{21} : Rotated hybrid composition function. (f) f_{24} : Rotated hybrid composition function.

improve on tuning the refreshing gap m [21]. The two biologically motivated approaches, GSO and ABC, completely fail in case of hybrid composition cases and suffer from the curse of dimensionality. On the other hand, SiPABC-Sf is propelled by the combined effect of two evolving mechanisms makes use of the saccadic flight strategy in executing intelligent search and diversifying forager locomotion through color guided selection

of locomotion pool. There are instances like wide optima basin (f_{18}) where the parameter adaptation may be crucial to find the optima. On the other hand, cases global optima being located in narrow basins (f_6 and f_{19}) calls for a local search optimizer that thoroughly scans the neighborhood for better solutions. The combined effect of both helps SiPABC-Sf to provide superior results in challenging situations (f_6 , f_{18} , and f_{19}).

Sample convergence characteristics corresponding to the median run of the peer algorithms (when the runs are ranked according to the best-of-the-run fitness values) have been shown in Fig. 7 for six test functions. All the plots indicate the better convergence speed of SiPABC-Sf as compared to the other algorithms.

D. Complexity Analysis and Computational Effort

For worst case complexity analysis we consider a D -dimensional problem with colony size SN . Considering that unit operations deal with the problem parameters, we can express the complexity in terms of SN and D . Initially each of the D parameters for SN trial solutions is initialized leading to $O(SN \cdot D)$. Next comes the CGL where the probability of modifying a problem parameter is MR . Worst case complexity will occur when all the D parameters for each trial solution is perturbed after traversing two stages. We assume that half of the problem's parameters get perturbed in CGL while other half in the next phase. Evidently from the pseudocode, the value of CS_i is generated (line 8), compared and saved (line 9) leading to three computations. This is followed by selecting x random members and one of the top $N\%$ members needed for perturbation thereby incurring $(x+1)$ computations. The by $D/2$ parameters are then perturbed and saved. Since $(x+4) < D/2$, with increasing dimensionality, the complexity of CGS can be better approximated as $O(SN \cdot D)$. As said earlier, the matrices are calculated using the properties of symmetry and skew-symmetry incurring $SN(SN - 1)/2$ computations. Same goes for calculating probability. The sorting based on fitness value has a complexity of $O(SN \cdot \log SN)$ since it is independent of dimensions. Thus, matrix and sorting operations can be approximated by a complexity of $O(SN^2 \cdot D)$. Now we move on to the saccadic flight strategy whereby the remaining $D/2$ parameters are perturbed using the correlation based model. Hypothetically if centering response is initiated for the entire colony irrespective of their locomotion recorded earlier, maximum complexity will be incurred due to the calculation of both $\theta_{bestfit}$ and $\theta_{worstfit}$. To do this, we proceed as follows. With respect to the i^{th} food source θ_i^C , we calculate the T/E ratio for $k \neq i$. Bookmarking helps to reduce complexity. We use two indicators to store the value of k for which the maximum and minimum value of $TE_{i,k}$ is obtained. These indicators store the index for $\theta_{bestfit}$ and $\theta_{worstfit}$, respectively. This leads to linear complexity $O(SN)$. The number of computations needed to calculate δ while instantiating centering, seeking or escaping response is of order x and can be safely ignored. Now, we proceed with perturbation of the $D/2$ parameters that were previously unaltered. As explained above, this leads to complexity of $O(D)$. Thus, for each food source there is approximately $\sim (SN + D)$ calculations. When problem dimensionality increases, i.e., $D > SN$, we can approximate this

as $O(SN.D)$. In *SiPABC-Sf*, the main computational time is incurred due to the distance calculations during matrix operation. The use of synchronous update does away with the need of repeated evaluation and updating of the affinity matrix for each replacement during greedy selection. We need to calculate the matrices only once per cycle. But the worst case is not the actual reflective of the algorithm's performance while dealing with real life engineering problems [32]. The empirical analysis has been presented in the supplementary attachment.

VI. CONCLUSION

The presented research has its source in the optic flow of information in flying insects and takes an initiative in the direction of multifaceted hybridization of the individual SI aspects, such collective intelligence, labor division, cooperation and many more. Achieving this goal requires the integration of the behavioral characteristics of foragers. The constant interpretation of geographical neighborhood is mapped into the objective space which aids the honeybees in locomotion. During locomotion, honeybees link their geo-spatial neighborhood to their associative learning and past memory while moving through intelligent search paths. The proposed *SiPABC-Sf* algorithm presents a novel foraging model by interlinking the evolving mechanisms involved in forager locomotion. The algorithmic model has been devised, based on the stimulus from the environment (search space), keeping close resemblance to the biological process. This novel, biologically motivated metaheuristics *SiPABC-Sf*, is an attempt to overcome the shortcomings of AE by incorporating the algorithmic analogues of our current understanding of biological process. To the best of our knowledge the *SiPABC-Sf* approach is the first of its kind that uses optic flow model in a swarm-based metaheuristic for real-parameter optimization. Through extensive experimentation and detailed analysis, we can infer that our biologically simulated process obtains significant predominance over highly competitive state-of-the-arts.

One of the most noticeable features in biological swarms is their ability to migrate from one place to another as a part of their evolving adaptation for survival. It will be interesting to develop a biological model closely resembling swarm multipopulation and simulate the migration tendency among them through cooperative coevolution.

REFERENCES

- [1] T. J. Carew "Associative learning in honeybees," *Behavioral Neurobiology: The Cellular Organization of Natural Behavior*. Sunderland, MA, USA: Sinauer Associates, 2000.
- [2] H. de Vries and J. C. Biesmeijer, "Modelling collective foraging by means of individual behavior rules in honey-bees," *Behav. Ecol. Sociobiol.*, vol. 44, no. 2, pp. 109–124, Nov. 1998.
- [3] T. Sato and M. Hagiwara, "Bee System: Finding solution by a concentrated search," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. Simulation*, vol. 4. Oct. 1997, pp. 3954–3959.
- [4] D. Karaboga, B. Gorkemli, C. Ozturk, and N. Karaboga, "A comprehensive survey: Artificial bee colony (ABC) algorithm and applications," *Artif. Intell. Rev.*, Springer, 2012, DOI: <http://10.1007/s10462-012-9328-0>
- [5] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," *Comput. Eng. Dept. Eng. Faculty, Erciyes University, Kayseri, Turkey, Tech. Rep. TR06*, 2005.
- [6] D. N. Lee, "The optic flow model: The foundation of vision," *Phil. Trans. Roy. Soc. London, Ser. B, Biol. Sci.*, vol. 290, no. 1038, pp. 169–178, Jul. 1980.
- [7] J. P. Lindemann, H. Weiss, R. Möller, and M. Egelhaaf, "Saccadic flight strategy facilitates collision avoidance: Closed-loop performance of a cyber-fly," *Biol. Cybern.*, vol. 98, no. 3, pp. 213–227, 2008.
- [8] R. F. A. Moritz and E. E. Southwick, *Bees as Superorganisms: An Evolutionary Reality*. Berlin, Germany/New York, NY, USA: Springer-Verlag, 1992.
- [9] N. R. Franks, S. C. Pratt, E. B. Mallon, N. F. Britton, and D. J. T. Sumpter, "Information flow, opinion polling and collective intelligence in house-hunting social insects," *Phil. Trans. Roy. Soc. London, Ser. B, Biol. Sci.*, vol. 357, no. 1427, pp. 1567–1583, Nov. 2002.
- [10] T. M. Judd, "The waggle dance of the honey bee: Which bees following a dancer successfully acquire the information?" *J. Insect Behav.*, vol. 8, no. 3, pp. 343–354, 1995.
- [11] R. Menzel and W. Backhaus, "Color vision in honey bees: Phenomena and physiological mechanisms," in *Facets of Vision*, D. Stavenga and R. Hardie, Eds. Berlin-Heidelberg, Germany/New York, NY, USA: Springer, 1989, pp. 281–297.
- [12] K. V. Frisch, *Bees: Their Vision, Chemical Senses, and Language*. Ithaca, NY, USA: Cornell Univ. Press, 1956.
- [13] M. V. Srinivasan, M. Lehrer, W. H. Kirchner, and S. W. Zhang, "Range perception through apparent image speed in freely flying honeybees," *Visual Neurosci.*, vol. 6, no. 5, pp. 519–535, 1991.
- [14] M. V. Srinivasan, M. Potesar, and K. Kral, "Motion detection in insect orientation and navigation," *Vision Res.*, vol. 39, pp. 2749–2766, Aug. 1999.
- [15] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm," *J. Glob. Optimiz.*, vol. 39, no. 3, pp. 459–471, 2007.
- [16] D. Karaboga and B. Basturk, "On the performance of artificial bee colony (ABC) algorithm," *Appl. Soft Comput.*, vol. 8, no. 1, pp. 687–697, 2008.
- [17] B. Akay and D. Karaboga, "A modified artificial bee colony algorithm for real-parameter optimization," *Inf. Sci.*, vol. 192, pp. 120–142, Jun. 2012.
- [18] S. Das and P. N. Suganthan, "Differential evolution: A survey of the state-of-the-art," *IEEE Trans. Evol. Comput.*, vol. 15, no. 1, pp. 4–31, Feb. 2011.
- [19] H.-G. Beyer and H.-P. Schwefel, "Evolution strategies: A comprehensive introduction," *Natural Comput.*, vol. 1, no. 1, pp. 3–52, 2002.
- [20] J. P. Lindemann, H. Weiss, R. Möller, and M. Egelhaaf, "Saccadic flight strategy facilitates collision avoidance: Closed-loop performance of a cyber-fly," *Biol. Cybern.*, vol. 98, no. 3, pp. 213–227, 2008.
- [21] S. Das, S. Biswas, and S. Kundu, "Synergizing fitness learning with proximity-based food source selection in artificial bee colony algorithm for numerical optimization," *Appl. Soft Comput.*, vol. 13, no. 12, pp. 4676–4694, 2013.
- [22] P. N. Suganthan, N. Hansen, J. J. Liang, K. Deb, Y.-P. Chen, A. Auger, et al., "Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization," Nanyang Technol. Univ., Nanyang, Singapore, May 2005 AND KanGAL Report #2005005, IIT Kanpur, India.
- [23] A. K. Qin, V. L. Huang, and P. N. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 398–417, Apr. 2009.
- [24] J. Zhang and A. C. Sanderson, "JADE: Adaptive differential evolution with optional external archive," *IEEE Trans. Evol. Comput.*, vol. 13, no. 5, pp. 945–958, Oct. 2009.
- [25] J. Brest, S. Greiner, B. Bošković, M. Mernik, and V. Žumer, "Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems," *IEEE Trans. Evol. Comput.*, vol. 10, no. 6, pp. 646–657, Dec. 2006.
- [26] M. S. Kiran and M. Gündüz, "A recombination-based hybridization of particle swarm optimization and artificial bee colony algorithm for continuous optimization problems," *Appl. Soft Comput.*, vol. 13, no. 4, pp. 2188–2203, Apr. 2013.
- [27] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions," *IEEE Trans. Evol. Comput.*, vol. 10, no. 3, pp. 281–295, Jun. 2006.
- [28] S. He, Q. H. Wu, and J. R. Sanders, "Group search optimizer: An optimization algorithm inspired by animal searching behaviour," *IEEE Trans. Evol. Comp.*, vol. 13, no. 5, pp. 973–991, Oct. 2009.

- [29] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics*, vol. 1, no. 6, pp. 80–83, Dec. 1945.
- [30] J. Derrac, S. García, D. Molina, and F. Herrera, "A practical tutorial on the use of non-parametric tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm Evol. Comput.*, vol. 15, no. 1, pp. 3–18, 2011.
- [31] S. García, A. Fernandez, J. Luengo, and F. Herrera, "Advanced non-parametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power," *Inf. Sci.*, vol. 180, no. 10, pp. 2044–2064, 2010.
- [32] Y. Jin, "A comprehensive survey of fitness approximation in evolutionary computation," *Soft Comput.*, vol. 9, no. 1, pp. 3–12, 2005.



Swagatam Das (M'10–SM'12) is currently an Assistant Professor with the Electronics and Communication Sciences Unit of Indian Statistical Institute, Kolkata, India. He has published one research monograph, one edited volume, and over 150 research articles in peer-reviewed journals and international conferences. His current research interests include evolutionary computing and pattern recognition.

He is the Founding Co-Editor-in-Chief of *Swarm and Evolutionary Computation*, an international journal from Elsevier. He serves as an Associate Editor of the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS, the IEEE ACCESS, *Neurocomputing, Information Sciences*, and *Engineering Applications of Artificial Intelligence*. He is an Editorial Board Member of *Progress in Artificial Intelligence* (Springer), *Mathematical Problems in Engineering*, *International Journal of Artificial Intelligence and Soft Computing*, and *International Journal of Adaptive and Autonomous Communication Systems*. He has been associated with the international program committees and organizing committees of several regular international conferences, including IEEE WCCI, IEEE SSCI, SEAL, GECCO, and SEMCCO. He has been a Guest Editor for special issues in journals such as the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, *ACM Transactions on Adaptive and Autonomous Systems*, and the IEEE TRANSACTIONS ON SYSTEM, MAN, AND CYBERNETICS, PART C: APPLICATIONS AND REVIEWS. He was the recipient of the 2012 Young Engineer Award from the Indian National Academy of Engineering.



Bijaya K. Panigrahi (SM'06) is an Associate Professor with the Department of Electrical Engineering, Indian Institute of Technology Delhi (IIT Delhi), Delhi, India. From 1992 to 2005, he was a Faculty Member with the Department of Electrical Engineering, UCE Burla, Odisha, India. His current research interests include application of soft computing and evolutionary computing techniques to power system planning, operation and control.

He has served as the Editorial Board Member/Associate Editor/Special Issue Guest Editor for different international journals. He is also associated with various international conferences in various capacities.



Souvik Kundu was born in W. Midnapore, West Bengal, India, in 1992. He is currently pursuing the B.E.Tel.E. degree from the Department of Electronics and Telecommunication Engineering, Jadavpur University, Kolkata, India.

He has published research articles in peer-reviewed journals and international conference proceedings. His current research interests include cyber-physical systems, analog circuits, and evolutionary computing.



Subhodip Biswas was born in Palta, West Bengal, India, in 1991. He is currently pursuing the B.E. degree from the Department of Electronics and Telecommunication Engineering, Jadavpur University, Kolkata, India.

He has published research articles in international conference proceedings and peer-reviewed journals. He has also reviewed papers for journals like *Information Sciences*, *Applied Soft Computing*, *Progress in Electromagnetics Research*, and *Neurocomputing*. His current research interests include machine learning, data mining, evolutionary computation, and their applications to various real-world problems, especially in bioinformatics and computational biology.



Debabrota Basu was born in Hooghly, West Bengal, India, in 1992. He is currently pursuing the B.E.Tel.E. degree from the Department of Electronics and Telecommunication Engineering, Jadavpur University, Kolkata, India.

He has published and presented research articles in international conference proceedings. His current research interests include developing computational geometry based and statistical tools for machine learning, computer vision, robotics and their application in several real world problems, especially in computational biology, neuroengineering, and biomedical imaging.