ON THE PERFORMANCE OF AN ARTIFICIAL BEE COLONY OPTIMIZATION ALGORITHM APPLIED TO THE ACCIDENT DIAGNOSIS IN A PWR NUCLEAR POWER PLANT

Ioná Maghali S. de Oliveira, Roberto Schirru and Jose A. C. C. de Medeiros

1 Universidade Federal do Rio de Janeiro
COPPE-Nuclear
Ilha do Fundão s/n, P. O. Box 68509
21941-972, Rio de Janeiro, Brazil
maghali@lmp.ufrj.br
schirru@lmp.ufrj.br
canedo@lmp.ufrj.br

ABSTRACT

The swarm-based algorithm described in this paper is a new search algorithm capable of locating good solutions efficiently and within a reasonable running time. The work presents a population-based search algorithm that mimics the food foraging behavior of honey bee swarms and can be regarded as belonging to the category of intelligent optimization tools. In its basic version, the algorithm performs a kind of random search combined with neighborhood search and can be used for solving multi-dimensional numeric problems. Following a description of the algorithm, this paper presents a new event classification system based exclusively on the ability of the algorithm to find the best centroid positions that correctly identifies an accident in a PWR nuclear power plant, thus maximizing the number of correct classification of transients. The simulation results show that the performance of the proposed algorithm is comparable to other population-based algorithms when applied to the same problem, with the advantage of employing fewer control parameters.

1. INTRODUCTION

It is a well known fact that classical optimization techniques impose several limitations on solving mathematical programming models. This is mainly due to inherent solution mechanisms of these techniques. Solution strategies of classical optimization algorithms are generally depended on the type of objective and variables used in the problem modeling. Their efficiency is also very much dependent on the size of the solution space, number of variables and constraints used in the problem, as well as the structure of the search space.

One of the main characteristics of the classical optimization algorithms is their inflexibility to adapt the solution algorithm to a given problem. This generally requires making several assumptions which might not be easy to validate in many situations. In order to overcome these limitations, more flexible and adaptable algorithms are needed, making it possible to model a given problem as close as possible to reality.

Based on this motivation, many nature inspired algorithms were developed in the literature such as Genetic Algorithm (GA) [1], Simulated Annealing (SA) [2] and Tabu Search (TS) [3].
It has also been shown that these algorithms can provide far better solutions in comparison to classical algorithms.

A branch of nature inspired algorithms which are called as swarm intelligence is focused on insect behavior in order to develop some meta-heuristics which can mimic insect’s problem solution abilities. Interactions between insects contribute to the collective intelligence of the social insect colonies and these interactions have been successfully adapted to scientific problems for optimization. Particle Swarm Optimization (PSO) [4] and Ant Colony Optimization (ACO) [5] are some of the well-known algorithms that mimic insect behavior in problem modeling and solution.

Observations and studies on honey bee behaviors resulted in a new generation of optimization algorithms. Versions for both combinatorial [6, 7, 8, 9, 10, 11, 12, 13, 14] and continuous [15, 16, 17] optimization problems have been presented in the literature. For optimizing multi-variable and multi-modal numerical functions, Karaboga has described the Artificial Bee Colony (ABC) Algorithm [17] in 2005.

The Artificial Bee Colony (ABC) algorithm is a relatively new member of swarm intelligence and tries to model the natural behavior of real honey bees in food foraging. Honey bees use several mechanisms to optimally locate food sources and to search new ones. This makes them good candidates for developing new intelligent search algorithms.

In order to illustrate the application of the ABC algorithm to complex problems in nuclear engineering, this work presents the utilization of the ABC algorithm as a tool for the development of a new event classification solution to help operators in the task of identifying possible plant transients in a nuclear power plant, thus making easier the decision-making process regarding to the plant safety in risk situations.

This paper is organized as follows: this section presents an overview of the work. Section 2 briefly discusses the transient classification problem. Section 3 describes the behavior of real bees and the basic Artificial Bee Colony (ABC) algorithm is introduced. Section 4 briefly explains the idea of the transient signatures of a PWR power plant that will be used as reference for comparisons with similar works in the literature. Section 5 gives the results obtained with this new approach. Section 6 presents some comparisons with similar works. Finally, Section 7 presents the conclusions of the present work.

2. TRANSIENTS CLASSIFICATION

The identification of possible transients in a nuclear power plant is a highly relevant problem. This is mainly due to the fact that the operation of a nuclear power plant involves a large number of variables whose behavior is extremely dynamic.

In risk situations, besides the huge cognitive overload that operators are submitted to, there is also the problem related with the considerable decrease in the effective time for correct decision making. To minimize these problems, transient identification systems have been developed in order to help the operators in the task of identifying possible transients in the plant and make the correct actions in due time.
A transient identification system may use a signature of a group of variables for each postulated transient of the plant. If a transient occurs, the transient identification system compares the evolution of the variables being monitored for this set of signatures and thus, the transient is classified as one of the postulated transients whose signature is closest (or more similar) to the ongoing transient, according to a given measure.

There are some ways to classify plant transients and each way depends on many factors. One way consists in comparing the Euclidian distances of the points of the event signatures with the centroids of the prototypes of the postulated transient signatures of the plant. The event is classified as the one whose distance of evolution of each variable to the prototype signature is the nearest, according to the Euclidian distance measure.

The objective of the minimization methods normally used is to try to find the optimal position of the prototypes that represent the centroids of the accident signatures in order to maximize the number of correct classifications. For a predetermined number of prototypes, the performance of each one is evaluated by a function that measures the number of correct classifications of the considered transients.

In this work, the task of finding the best prototypes that correctly identifies three different types of postulated transients for a PWR nuclear power plant was attributed to the ABC algorithm, which is described in the next Section.

Detailed information about the transient identification problem may be seen in [18].

3. ARTIFICIAL BEE COLONY (ABC) ALGORITHM

3.1. Minimal Model of Foraging Behavior of Honey Bees

Social insect colonies can be considered as dynamical system gathering information from the environment and adjusting its behavior in accordance to it. While gathering information and adjustment processes, individual insects do not perform all the tasks because of their specializations. Generally, all social insect colonies behave according to their own division labours.

The model that leads to the emergence of collective intelligence of honey bee swarms consists of three essential components: food sources, employed foragers and unemployed foragers, and defines two leading modes of the behavior: recruitment to a nectar source and abandonment of a source [17].

i) Food sources: the value of a food source depends on many factors, such as its proximity to the nest, richness or concentration of energy and the ease of extracting this energy. For the simplicity, the “profitability” of a food source can be represented with a single quantity.

ii) Employed foragers: they are associated with a particular food source, which they are currently exploiting or are “employed” at. They carry with them information about this particular source as its distance, direction from the nest and the
profitability of the source and they share this information with a certain probability.

iii) Unemployed foragers: they are looking for a food source to exploit. There are two types of unemployed foragers: scouts (searching the environment surrounding the nest for new food sources) and onlookers (waiting in the nest and finding a food source through the information shared by employed foragers).

The exchange of information among bees is the most interesting occurrence in the formation of collective knowledge. The most important part of the hive with respect to exchanging information is the dancing area. Communications among bees related to the quality of food is called waggle dance. Since information about all the current rich sources is available to an onlooker on the dance floor, she probably could watch numerous dances and choose to employ herself at the most profitable source. There is a greater probability of onlookers choosing more profitable sources since more information is circulating about the more profitable sources. Employed foragers share their information with a probability, which is proportional to the profitability of the food source, and the sharing of this information through waggle dancing is longer in duration. Hence, the recruitment is proportional to profitability of a food source.

3.2. The Basic Artificial Bee Colony (ABC) Algorithm

The Artificial Bee Colony (ABC) algorithm, proposed by Karaboga for optimizing numerical problems in [17], simulates the intelligent foraging behavior of honey bee swarms. In ABC algorithm, the colony of artificial bees contains three groups of bees: employed bees, onlookers and scouts. A bee waiting on the dance area for making decision to choose a food source is called an onlooker and a bee going to the food source visited by itself previously is named an employed bee. A bee carrying out random search is called a scout. In the ABC algorithm, first half of the colony consists of employed artificial bees and the second half constitutes the onlookers. The number of employed bees is equal to the number of food sources. The employed bee whose food source has been exhausted becomes a scout bee.

The search carried out by the artificial bees can be summarized as follows:

- Each employed bee determines a food source within the neighborhood of the food source in her memory and evaluates its profitability.
- Each employed bee shares information with onlookers waiting in the hive and then each onlooker selects a food source site depending on the information given.
- Each onlooker determines a food source within the selected site chosen by herself and evaluates its profitability.
- An employed bee of which the source has been abandoned becomes a scout and starts to search a new food source randomly.

In the ABC algorithm, the position of a food source represents a possible solution to the optimizing problem and the nectar amount of a food source corresponds to the quality
fitness) of the associated solution. The number of the employed bees is equal to the number of food sources being exploited at the moment or to the number of solutions in the population.

The main steps of the algorithm are given below:

1. Initialize the population of solutions $x_{i,j}$
2. Evaluate the population
3. cycle = 1
4. repeat
5. Produce new solutions (food source positions) $v_{i,j}$ in the neighborhood of $x_{i,j}$ for the employed bees using the formula $v_{i,j} = x_{i,j} + \phi_i(x_{i,j} - x_{k,j})$ (k is a random solution in the neighborhood of $i$, $\phi$ is a random number in the range [-1,1]) and evaluate them
6. Apply the greedy selection process between $x_i$ and $v_i$
7. Calculate the probability values $P_i$ for the solutions $x_i$ by means of their fitness values using the equation (1):

$$P_i = \frac{fit_i}{\sum_{i=1}^{SN} fit_i}$$  \hspace{1cm} (1)

In order to calculate the fitness values of solutions we employed the equation (2):

$$fit_i = \begin{cases} 
\frac{1}{1 + f_i} & \text{if } f_i \geq 0 \\
1 + \text{abs}(f_i) & \text{if } f_i < 0
\end{cases}$$  \hspace{1cm} (2)

Normalize $P_i$ values into [0,1]

8. Produce the new solutions (new positions) $v_i$ for the onlookers from the solutions $x_i$ selected depending on $P_i$ and evaluate them
9. Apply the greedy selection process for the onlookers between $x_i$ and $v_i$
10. Determine the abandoned solution (source), if exists, and replace it with a new randomly produced solution $x_i$ for the scout using the equation (3):

$$x_{i,j} = \min_j + \text{rand} (0,1)*(\max_j - \min_j)$$  \hspace{1cm} (3)

11. Memorize the best food source position (solution) achieved so far
12. cycle = cycle + 1
13. until cycle = Maximum Cycle Number (MCN)

The nectar amount of a food source corresponds to the quality of the solution represented by that food source. Onlookers are placed onto the food sources by using “roulette wheel selection” method [1]. Every bee colony has scouts that are the colony’s explores. The explorers do not have any guidance while looking for food. They are primarily concerned with finding any kind of food source. As a result of such behavior, the scouts are
characterized by low search costs and a low average in food source quality. Occasionally, the scouts may accidentally discover rich entirely unknown food sources. In the case of artificial bees, the artificial scouts might have the fast discovery of the group of feasible solutions as a task. In ABC algorithm, one of the employed bees whose food source has been exhausted is selected and classified as the scout bee. The classification is controlled by a control parameter called \textit{limit}. If a solution representing a food source is not improved until a predetermined number of trials, then that food source is abandoned by its employed bee and the employed bee becomes a scout. The number of trials for releasing a food source is equal to the value of \textit{limit}, which is an important control parameter of ABC algorithm.

Common to all population-based search methods is a strategy that generates variations of the solution being sought. Some search methods use a greedy criterion (greedy solution) to decide which generated solution to retain. Such a criterion would mean accepting a new solution if and only if it increases the value of the objective function.

Except common parameters (population number and maximum evaluation number), the basic ABC used in this paper employs only one control parameter which is called \textit{limit}. We defined a relation for the \textit{limit} value by using the dimension of the problem and the colony size as follows:

\[ \text{limit} = SN \times D \]

Where \( D \) is the dimension of the problem and \( SN \) is the number of food sources or employed bees.

4. NUCLEAR POWER PLANT TRANSIENTS SIGNATURES

In order to identify abnormal events, a transient diagnosis support system for a nuclear power plant may use a set of signatures of some plant variables, measured for each abnormal event of the plant. A list with some postulated transients as well as a list of some process variables that best contribute to the characterizations of these events may be seen in [18,19].

When an abnormal operation event needs to be identified, the diagnostic system compares the evolution of the measured plant variables with the signatures of the evolution of these variables for each abnormal postulated event of the plant.

In this work, instead of the three partitions considered in [18], we considered only one partition and the evolution of the associated variables for each one of the 3 postulated accidents listed in Table 3 for a nuclear power plant operating at 100\% at nominal power in order to verify the efficiency of the ABC algorithm in determining the optimal position of the prototypes that represent the centroids of the accident signatures, maximizing the number of correct classifications.

<table>
<thead>
<tr>
<th></th>
<th>BLACKOUT</th>
<th>Loss of external electric power</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>LOCA</td>
<td>Loss of coolant accident</td>
</tr>
<tr>
<td>3</td>
<td>SGTR</td>
<td>Rupture of steam generator tubes</td>
</tr>
</tbody>
</table>

Table 1. Numerical results to the problem model

INAC 2009, Rio de Janeiro, RJ, Brazil.
According to [18], there are 18 variables associated with each one of the 3 transients considered in this work (see Table 3). So, the number of prototypes is thus calculated: \((NV + 1) \times NT\), where NV is the number of variables associated with each transient, 1 is the time unit and NT is the number of transients. This way, to our problem we have: \((18 + 1) \times 3 = 57\), which is the dimension \(D\) considered for the problem.

To evaluate the fitness of each prototype, we used an objective function that weights favorably the number of the correct classifications. The fitness is characterized by the difference of the maximum number of classifications and the number of correct classifications.

5. RESULTS

The results obtained with the application of the ABC algorithm to the problem described in the previous sections are given in Table 2. For the implementation of the algorithm, use was made of the MATLAB environment. The experiments considered different colony sizes as well as different random seeds and the number of cycles was fixed in 10,000 for each run. The dimension of the problem was defined as \((NV + 1) \times NT\), where NV is the number of variables and NT is the number of transients.

As one can see from Table 2, the ABC algorithm was able to obtain a number bigger than 95% of correct classifications using a lesser number of parameters than other population-based algorithms. In the next section we compare the performance of the ABC algorithm with the performance of GA, PSO and ACO algorithms applied to the same problem.

Figure 1(a) shows the fitness evolution over 10,000 generations. Figure 2(b) shows the performance of ABC algorithm related to the problem in question. \(A = 177\) corresponds to the number of correct classifications (96.7%) while \(F = 006\) represents the number of incorrect classifications. \(NC = 1\) corresponds to the number of partitions, that in this paper was defined to be one.

![Table 2. Parameter values used in the experiments](image)

<table>
<thead>
<tr>
<th>colony size</th>
<th>100</th>
<th>140</th>
<th>240</th>
<th>300</th>
<th>360</th>
</tr>
</thead>
<tbody>
<tr>
<td>ne</td>
<td>50</td>
<td>70</td>
<td>120</td>
<td>150</td>
<td>180</td>
</tr>
<tr>
<td>no</td>
<td>50</td>
<td>70</td>
<td>120</td>
<td>150</td>
<td>180</td>
</tr>
<tr>
<td>limit</td>
<td>(ne \times D)</td>
<td>(ne \times D)</td>
<td>(ne \times D)</td>
<td>(ne \times D)</td>
<td>(ne \times D)</td>
</tr>
<tr>
<td>cycles</td>
<td>10,000</td>
<td>10,000</td>
<td>10,000</td>
<td>10,000</td>
<td>10,000</td>
</tr>
<tr>
<td>correct classifications</td>
<td>177 (96.7%)</td>
<td>177 (96.7%)</td>
<td>177 (96.7%)</td>
<td>177 (96.7%)</td>
<td>177 (96.7%)</td>
</tr>
</tbody>
</table>

\(ne =\) number of employed bees; \(no =\) number of onlooker bees; \(D =\) dimension of the problem.
6. COMPARISONS WITH GA AND PSO ON THE IDENTIFICATION OF NUCLEAR POWER PLANT TRANSIENTS

In order to compare the results obtained with the ABC algorithm with the results obtained with GA and PSO in previous works [18, 20], in this paper, use was made of the MATLAB environment. For both GA and PSO algorithms, 10 tests were performed with different seeds for 100 generations of populations containing 500 individuals. However, for the ABC algorithm, a population of 500 individuals and a number of 100 generations proved to be inadequate. Using the data contained in Table 2, the ABC algorithm obtained results compatible with GA and PSO.

Table 3 shows the comparative results obtained with the application of GA, PSO and ABC algorithm for the problem of identification of transients using the selected set of transients of the Table 1.

<table>
<thead>
<tr>
<th>Maximum Number of Correct Classifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABC</td>
</tr>
<tr>
<td>GA</td>
</tr>
<tr>
<td>PSO</td>
</tr>
</tbody>
</table>

The evolution of the GA and PSO for this problem is illustrated in Figure 2(a) and 2(b) that presents the graphic of the fitness convergence for both algorithms.

Considering that, in the ABC algorithm each generation corresponds to 80 evaluations in the test case, it can be seen through the analysis of Figure 2 that the ABC performance is comparable to both GA and PSO.
7. CONCLUSIONS

The method of solution proposed in this work is a search-based that looks for the best prototype positions that represent the centroids of the accident signatures, maximizing the number of correct classifications. The easy way to implement and the small number of control parameters makes the ABC algorithm an interesting option of optimization technique. It also can be used in high dimension and complex multimodal search spaces, as in this work in which the ABC algorithm was used to implement a search-based method for the problem of transient classification.

The solutions obtained by the proposed method showed results similar to those obtained by similar methods [19, 20]. In particular, in the identification problem of three of the postulated transients, the method allowed a solution that approaches the ideal solution (the Voronoi vectors) for the classification of transients considering fewer partitions than those referred in literature.

Although the performance of the ABC algorithm can be improved by integrating useful heuristics, it is interesting to observe that there are some similarities between ABC and other population-based algorithms such as GA [1] and DE [21]. While GA and DE employ crossover operators to produce new or candidate solutions from the present ones, ABC algorithm does not. ABC algorithm produces the candidate solution from its parent by a simple operation based on taking the difference of randomly determined parts of the parent and a randomly chosen solution from the population. This process increases the convergence speed of search into a local minimum. In GA, DE and PSO the best solution found so far is always kept in the population and it can be used for producing new solutions in the case of DE and GA, new velocities in PSO [4]. However, in ABC, the best solution discovered so far is not always held in the population since it might be replaced with a randomly produced solution by a scout. Therefore, it might not contribute to the production of trial solutions. Both DE and ABC employ a greedy selection process between the candidate and the parent solutions. In ABC, on “employed bees” stage a trial solution is produced for each solution in the population as in the case of DE without depending on the quality of solutions. On
“onlooker bees” stage, the solutions with the higher fitness value are used more often than those with less fitness values to produce trial solutions. It means that the promising regions of the search space are searched in shorter time and in detail. This selection is similar to the natural selection or seeded selection employed in GA.

In GA or DE, mutation process creates a modification on a randomly selected part of a solution to provide required diversity in the population. In ABC, there are two type mechanisms to control diversity in the population: a) As in DE or GA, a randomly chosen part of a parent is modified with a magnitude determined randomly to obtain a trail solution. This modification is relatively small and useful for local search and fine tuning; b) Rather than changing a part of a solution, a whole solution in the population is removed and then a new one produced randomly is inserted into the population by a scout. This mechanism provides the ABC algorithm a global search ability and prevents the search from premature convergence problem. This feature weakens the dependency of the algorithms’ performance on the population size, too. Hence, there is a good balance between the local search process carried out by artificial onlooker and employed bees and global search process managed by artificial scouts.

Apart from the maximum evaluation number and population size, a standard GA has three more control parameters (crossover rate, mutation rate, generation gap), a standard DE has at least two control parameters (crossover rate, scaling factor) and a basic PSO has three control parameters (cognitive and social factors, inertia weight). Also, limit values for the velocities $v_{\text{max}}$ have a significant effect on the performance of PSO. The ABC algorithm has only one control parameter ($\text{limit}$) apart from Colony Size and Maximum Cycle Number. The expression for determining the value of $\text{limit}$ depends on the population (colony size) and dimension of the problem. Therefore, ABC has only two common control parameters: Maximum Cycle Number (MCN) and colony size (SN). Consequently, ABC is as simple and flexible as DE and PSO, also employs less control parameters, and shown to be efficient and robust for solving complex nuclear engineering problems.

ACKNOWLEDGMENTS

The authors are sponsored by Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq and Fundação de Amparo à Pesquisa do Estado do Rio de Janeiro – FAPERJ.

REFERENCES


