IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 15, 2020, accepted September 27, 2020, date of publication October 9, 2020, date of current version October 22, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3029890

Gene Selection Using Hybrid Multi-Objective Cuckoo Search Algorithm With Evolutionary Operators for Cancer Microarray Data

MOHD SHAHIZAN OTHMAN, SHAMINI RAJA KUMARAN, AND LIZAWATI MI YUSUF
School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia, Skudai 81310, Malaysia

Corresponding author: Mohd Shahizan Othman (shahizan@utm.my)

This work was supported in part by the School of Graduate Studies, Universiti Teknologi Malaysia, (Zamalah Scholarship), and in part by the Fundamental Research Grant Scheme (FRGS) under Grant 5F207.

ABSTRACT Microarray data play a huge role in recognizing a proper cancer diagnosis and classification. In most microarray data set consist of thousands of genes, but the majority number of genes are irrelevant to the diseases. An efficient algorithm for gene selection becomes important to deal with large microarray data. The main challenge is to analyze and select the relevant genes with maximum classification accuracy. Various algorithms were proposed for gene classification in previous studies, however, limited success was succeeded due to the selection of many genes in the high-dimensional microarray data. This study proposed and developed a hybrid multi-objective cuckoo search with evolutionary operators for gene selection. Evolutionary operators that are used in this article were double mutation and single crossover operators. The motivation behind this research is to improve the dimensions' values and explorative search abilities. Multi-objective cuckoo search with evolutionary operators employed the selection of informative genes among the high-dimensional cancer microarray data. Experiments were conducted on seven publicly available and high-dimensional cancer microarray data sets. These microarray data sets consist of approximately 2000 to 15000 genes. The results from the experiments concluded that the developed algorithm, multi-objective cuckoo search with evolutionary operators outperforms cuckoo search and multi-objective cuckoo search algorithms with a smaller number of selected significant genes.

INDEX TERMS Gene selection, cancer microarray data, cuckoo search, multi-objective, evolutionary operators.

I. INTRODUCTION

Cancer research is one of the active research fields in the medical areas and it has been ongoing for centuries. Research for cancer causes involves many different types of disciplines. Concerning the investigation of causes and potential treatment purposes, many biological microarray experiments have been conducted as an initial step solely to gain more information. Prompt identification of cancer is crucial since it is usually more complex to treat patients in later stages. Accurate prediction of cancer is significant in contributing effective treatment for the patients [1]. However, there are difficulties in detecting cancer because of a large amount of gene expression levels in the human body. Gene expression levels are known to have important keys to inscribe the fundamental problems related to the cure and prevention of diseases [2].

Microarray technology is introduced to define the global view of the cellular function of a gene by gene approaches. In addition to it, microarray technology is used to measure the gene expression's activity from the complete genome into one experiment [3]. Through microarray experiments, the investigation of genetic mechanisms of cancer leads toward discovering advanced drug designs in the medical industry [4]. In recent years, microarrays experiments have become famous because they held thousands of spots of different deoxyribonucleic acid (DNA) sequences and interrogate each gene in an organism. Microarray technology makes this experiment possible and the data generated from the experiments are enormous. These enormous amounts of data give representations and will provide information based on

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

computational methods that able to derive meaningful and significant results from the experiment [5]. The scientific tasks in microarray experiments involved analyzing microarray gene expression data which include the identification of co-expressed genes and sample discovery with similar expression patterns that are highly discriminative for discerned biological samples. Analyzing gene expression data from microarray technology is the current challenge encountered by researchers.

The tools that are used by the researchers to analyze large quantities of data were machine learning and data mining. In machine learning and data mining, classification is an important task that can classify an instance into the corresponding classes [6]. Each instance is described by features sets and class labels. The input sets of features are the key factor that influences the quality of the performance of a classification algorithm [7]. If the features are relevant to the class labels, the classifier able to generate a strong relationship between them. However, in most scenarios, the relevancy of features is often unknown and usually, the input data sets have issues such as irrelevancy and redundancy that are not useful during the knowledge discovery process [8]. Thus, this can hinder the process of producing a positive classification. The majority of real-world classification problems require knowledge on relevant features and reduction of irrelevant and redundant features can drastically reduce the size of data of the learning algorithms.

One of the prominent solutions for dimensionality reduction is feature selection or also known as gene selection under the context of this research. Gene selection is a process to reduce the dimensionality of data to enhance the recognition outcome. There are three mains steps for feature selection; search procedure, evaluation function, and stopping criterion. Along with the gene selection algorithm, other algorithms are required to reduce the complexity in gene space and also to identify the highly distinctive genes [9]. Due to the proliferation of high-dimensional features and data, it is difficult to extract the right information.

High-dimensional microarray data sets are difficult to interpret and interpretation of data is very important for the treatment of patient conditions. From the initial studies of dimensionalities, [10] found that the best test error can be attained through a limited number of features that directly affect the accuracy rates. In a large feature space, it is common to have irrelevant and redundant genes concerning the class labels. Integrality constraints such as irrelevant and redundant features can affect the classification performances. Therefore, this research study developed a gene selection algorithm to counter all the mentioned drawbacks. This is to produce an optimal feature space with significant genes that can produce better classification performance accuracy.

Inspired by previous researchers, metaheuristics algorithms are more suitable to optimize large and complex data. Techniques comprised of meta-heuristic optimization has a broad range from the process of a local search to learning processes. Metaheuristic by conducting them over the search space thereby bringing out its best capabilities able to obtain the best of best solutions [11]. Besides, metaheuristic algorithms include an evolutionary algorithm (EA) and swarm intelligence (SI) algorithms that becoming powerful and strong methods for solving many tough problems. Thus, this article is motivated to focus on the cuckoo search (CS) as the metaheuristic algorithm in solving the existing bottlenecks in a gene selection process. This is because CS is known for its efficiency as a swarm-intelligence based algorithm and instead of building a new metaheuristic algorithm, improvising the existing algorithm allows the efficiency of an algorithm to be enhanced.

The remainder of this article is organized as follows: Section II, related works based on gene selection using microarray data sets are discussed. The research methodology is provided in Section III. Section IV described the details of the gene selection algorithm, while Section V discussed the experimental design, following Section V1 until IX discusses the results and findings in-detailed. Finally, Section X presented the conclusion.

II. RELATED WORKS

Recently metaheuristic algorithms are famous in handling gene selection problems and the performance of the proposed techniques has been proved to show better performances. Even though there are many methods have been proposed for gene selection, however, most of them suffer from stagnation issues in local optima and high computational cost, thus, this cannot guarantee the optimality and relevancy of the identified genes from the use of metaheuristic algorithms in large search space [12], [13].

The authors in [14] present a qualitative mutual information (QMI) method for feature selection. Random forest's importance score is calculated in QMI, whereby these scores separate the correlated features and reduce the redundancies between genes. Nevertheless, to segregate the genes in the data which are irrelevant, the class label mutual information (MI) is utilized. MI helps to gain of each feature with the class variable. However, using a random forest to calculate the preference score is time-consuming and over-fitting. The developed method by [14] will be biased to certain features due to existing noise in the data.

Next, in [15] proposed an enhanced cuckoo search (ECS) and compared it with cuckoo search and harmony search for the classification of mammogram image-based breast cancer data sets. ECS is based on egg-laying behavior and the existence of multiple eggs in the nest. ECS handles a single constraint by enhancing the number of eggs as solutions in the scenario. However, there are more than one constraint need to be improved in cuckoo search so that the algorithm functions more effectively. But, the article [15] showed the credibility of the cuckoo search only as a feature algorithm with minimum features, it has performed with an average accuracy of 99.13% of the data using a kNN classifier.

The most popular and common algorithm for feature selection is particle swarm optimization (PSO) and recently,

it has been proposed by [16]–[18] for microarray data. Reference [16] used PSO to gain best-fit features. PSO explores the feature space and the reduced number of features to select significant features in breast cancer data. While [19] used Pearson's correlation coefficient (PCC) integrated with binary particle swarm optimization (BPSO) and compared with PCC and genetic algorithm (GA). Reference [17] proved BPSO has a better performance compared to GA for feature selection. Here, BPSO is introduced for that research in order to handle discrete variables.

While the authors in [18] used multi-population particle swarm optimization (MPSO) for feature selection to identify significant genes for two publicly available microarray data sets. MPSO is developed to enhance the search compared to general PSO. This is due to the drawback in PSO that tends to fall into local optima trap and incapability to explore for more assorted solutions.

Other than PSO, [19] proposed a multi-objective artificial bee colony algorithm to select the best genes for continuous optimization problems in a binary solution space. An artificial bee colony (ABC) imitates the bees' behavior and the bees are to maximize the number of nectar sources, while, minimizing the distance of the sources. As discussed by [19], feature selection is defined as a multi-objective optimization problem and increases the accuracy rates through multi-objective optimization. From [20] comparing ABC and integrated cuckoo search (CS) and ABC (CS-ABC), CS-ABC outperformed ABC with better accuracy results. The disadvantage in ABC is it suffers from premature convergence, thus, CS assist ABC to replace the not-so-good solutions into good solutions. Other than that, [21] has used two archive guided multi-objective artificial bee colony algorithm for cost-sensitive feature selection. Two archives which are leader and external archives are utilized to enhance the search capability of the algorithm. However, comparing to CS from this research with a bee colony, CS has a larger search ability due to its nature itself by focusing on the survival of best eggs or solutions. Thus, by having two archives using the bee colony computationally it will be a complex task.

The next recent research that focused on multi-objective criteria for feature selection was research by [22]. Reference [22] has proposed a binary differential evolution algorithm with a self-learning strategy to solve multi-objective feature selection problems. One of the research ideas by [22] is implementing three operators for multi-objective feature selection which were mutation operator, one-bit purifying search operator (OPS), and non-dominated sorting operator using binary differential evolution with self-learning. However, a non-dominated sorting operator is an element in Pareto archive to handle crowding distance which is also utilized in this research. Therefore, a drawback in [22] is required implementation of Pareto optimal and it does not require an additional operator such as OPS. While OPS is an operator with self-learning capability is a great advantage for future work in feature selection methods.

The advantage of CS is due to its global search using Lévy flights rather than standard random walks. Nevertheless, Lévy flights have infinite mean and variances whereby CS able to explore the search space more efficiently compared to other algorithms. The reason CS is implemented in different fields is due to its simple structure. Many researchers attempted to improve their efficiency to obtain a better solution to different problems. Reference [23] inspired to utilize multi-objective cuckoo search for gene selection instead only for optimization. This research would like to choose multi-objective optimization with cuckoo search (MOCS) because this research article has more than one objective to be achieved during the selection of best genes in cancer microarray data.

Furthermore, the idea of using evolutionary operators (EO) for gene selection with a metaheuristic algorithm emerged based on researches [24], [25]. These researches have highlighted the benefits of the classic operators that can assist in classification processes. EO such as crossover and mutation operators are the basis of the genetic algorithm's evolution. Crossover probabilistic selects two chromosomes from the current population-based on fitness values and combines to produce offspring [26]. While mutation operator ensures the population against permanent fixation by flipping the value of the bits of selected chromosomes in randomly selection positions [26]. The application of such operators in a search space able to improve the classification accuracy rates and speed up the process of searching [24]. Therefore, the capability of EO is an element that is unavoidable because it can boost the advantages in MOCS and can select the best genes. The functionalities of mutation and crossover operators are discussed as follows [27]:

(a) Mutation operator: Creates a new solution through one and only one evolving population member in each mutation event. The mutation operator randomly modifies one or more genes of a chromosome with a given probability with increasing the structural diversity of the diversity.
(b) Crossover operator: Involves more than one evolving member (parents) in creating a new solution (child) in each crossover event. The crossover combines the genes of two or more parents to produce a better child. This idea is based on the exchange of information between good chromosomes will generate even better child.

III. PROBLEM FORMULATION

Metaheuristic algorithms are nature-inspired and act as a strong tool in solving various problems. Cuckoo search (CS) is developed based on cuckoo birds' reproduction. Basically, the cuckoo breeding behavior is based on brood parasitism whereby the cuckoo lay one or many eggs in other birds' or hosts' nests. The aim of these cuckoo birds to do so is to ensure the continuity of their generation by leaving the host birds guided by their natural instinct of breed, hatch, and provide food to the baby cuckoos [28]. Cuckoos are smart

because to increase the chance of survival of its eggs, it gobbles an egg in the host's nests. Unfortunately, if the host of the nest identified that there is an alien egg in the nest, the host might throw these alien eggs or abandon the current nest and build a new nest [29], [30]. Hence, this imitation reduces the probability of cuckoo's eggs being thrown or abandoned. This might increase the survival and reproductive capacity of cuckoo birds. There are three idealized rules for CS [31]:

(a) A cuckoo lays one egg at a time and dumps its egg randomly chosen nest.
(b) The best nest with the high quality of eggs will carry over to the next generations.
(c) The availability number of host nests is secured and the egg laid by the cuckoo is detected with the probability, $p_a \in [1, 0]$. While the host bird either throw the egg or abandon the nest and build a new nest. The final assumption can be approximated by the fraction p_a of the n nests are replaced with new nests with randomized results.

The final rule can be approximated by replacing a fraction p_a of n hosts nests with new random nests (solutions). The fitness of a solution can be proportional to the values of objective functions. The simple representation of CS for this research, eggs in a nest acts as the solution and the cuckoo eggs represent the new solution. Furthermore, the objective here is to utilize the new and better solutions (cuckoo's eggs) to replace the poor solution in the nests. The algorithm might be complicated if a nest has different eggs represent a set of results. Algorithm 1 shows the basic procedures of CS [31]:

There are two main important elements have been implemented in the development of CS which are as follows:

(a) The brood parasitism of cuckoo.
(b) The breeding behavior of cuckoos with the Lévy flights' characteristics.

The typical behavior of Lévy flights was demonstrated by many researchers, for example, Lévy flights have been used in the firefly algorithm [32], particle swarm optimization (PSO) [33], and movement of hunters [30]. Lévy flight plays an important role in CS due to the pattern of free search. During the generation of new solutions, x^{t+1} and cuckoo i, a Lévy flight is performed [34]:

$$x_i^{(t+1)} = x_i^t + \propto \oplus \text{Levy}(\lambda) \quad (1)$$

where $\alpha > 0$ is related to the scales of interest's problems and $\oplus$ is the entrywise multiplication. In most of the cases, the researcher suggested using, $\alpha = 1$.

Equation (1) is essential for a random walk and it is a Markov Chain product that depicts the next location depends on the current location [35]. $\oplus$ is the product of entrywise whereby more efficient in exploring the search space (step length longer in the long run) via the Lévy flight. The random walk is provided by the Lévy flight while the length of random steps is drawn from a Lévy distribution [29], [36]. However, there are bottlenecks in CS. Therefore, the drawback in CS has been presented in Figure 1 adapted from [30].

Algorithm 1 Original Cuckoo Search Algorithm

```
Begin
Objective function f(x), x = (x_i, ..., x_d)^T
    Generate an initial population of n
    host nests, x_i (i = 1, 2, ..., n)
      while (t < MaxGeneration) or
      (stop criterion)
        Get a cuckoo randomly by Lévy
        flights
        Evaluate its quality/fitness F_i
        Choose a nest among n (say j)
        randomly
        if (F_i > F_j)
         i^th the new solution
        end if
        A fraction (p_a) of worse nests
        are abandoned and new
        are built;
        Keep the nests with best
        quality solutions;
        Rank the solution and find
        the current best;
      end while
    Post process results and
    visualization
End
```

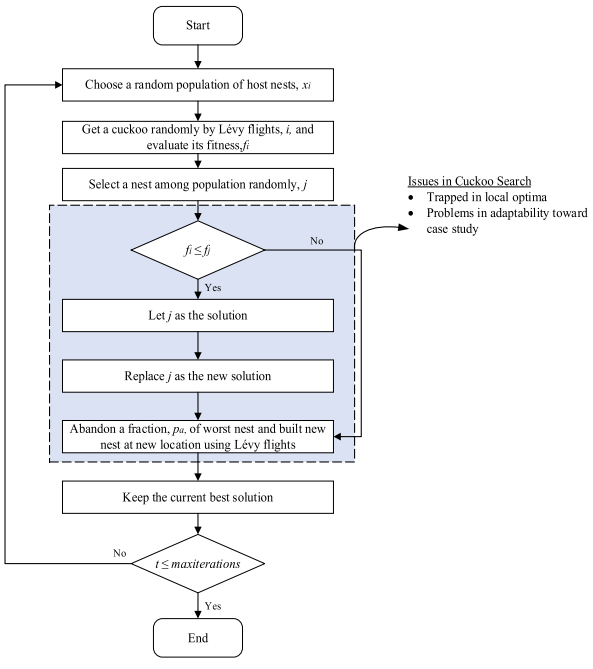


FIGURE 1. Drawbacks in cuckoo search.

The drawbacks in CS are easily trapped in local optima and difficult to obtain ideal search results because hard to adapt based on the case study [37]. Trapping in local optimum is falling into an extreme point in the iterative search process and in sudden it ends the operation. While the adaptability

issues are due to fixed parameter values which might lower the chance to get an optimal solution. In order to overcome the drawbacks, the next section discussed the proposed solution for gene selection.

IV. HYBRID MULTI-OBJECTIVE CUCKOO SEARCH-EVOLUTIONARY OPERATORS (MOCS-EO)

In this section, this article will present the details of the algorithms. A single objective CS would like to improve performance measures such as accuracy rates, however, in this research, two main contradict objectives to be achieved using cuckoo search. The multi-objectives that being achieved through the cuckoo search were as follows:

(a) Maximize the classification measures to improve the decision-making using cancer microarray data.
(b) Minimize the number of genes to improve the relevancy and reduce the redundancy of cancer microarray data.

Gene selection (GS) is the possible set of genes in a data and ψ is the set of objective functions that are required for multi-objective optimization. While *optimal* refers to either maximum or minimum needs that depend on the objective's nature.

$$Optimize\ \psi(GS) = Optimal|g\_1(GS), g\_2(GS)\ldots, g\_N(GS)| \quad (2)$$

Thus, the following equations refer to objectives that deal with maximization and minimization problems,

$$Maximize\ \psi(GS) = Max|g_1(GS), g_2(GS)\ldots, g_N(GS)| \quad (3)$$

$$Minimize\ \psi(GS) = Min|g_1(GS), g_2(GS)\ldots, g_N(GS)| \quad (4)$$

It is difficult to optimize one objective without affecting the other objective. Therefore, Pareto optimal solution is used as the set of the optimum solution with varying degrees of objective values. The aim here is to identify the Pareto optimal that able to include all solutions in the search space and also comprises the non-dominating solutions within the search space. In this scenario, the stated two objectives are considered equally important even though it contradict each other.

Multi-objective cuckoo search (MOCS) is required to sort the population based on the ascending level of non-domination. This non-domination process repeats until all solutions that are chosen are best and sorted. The crowding distance metric is an important component of this algorithm. The aim behind using the crowding distance metric is to bring a set of diverse solutions. Crowding distance is the density estimator for the solutions in a population. The crowding distance of a solution is formulated based on $i + 1$ and i-1 (the average distance of each solution along with each objective).

To deal with multi-objective optimization, the first step is the parameter settings. For MOCS, there are two main parameters to be initialized which are population size, N, and a fraction, p_a of the worst nests to replaces or might be rejected. In the second step, a swarm of N host nests denotes as the possible solutions. Thus to provide good initial solutions, this research utilized Pareto optimal. The complete algorithm of multi-objective cuckoo search with evolutionary operators (MOCS-EO) algorithms is as shown in Algorithm 2.

Algorithm 2 Proposed MOCS-EO Algorithm

```
INPUT: Multi-objective Cuckoo Search
Parameters
OUTPUT: Selected significant genes
Begin
Objective function f(x), x = x_1,...,x_d)^T
Generate initial population of n host
nests with m eggs using mutation
operator (Pm)
while (t < MaxGeneration) or (stop the
criterion)
   for each nest
      Get a cuckoo randomly (assume i)
      by Lévy flights
        Check the eggs
          if eggs = cuckoo_eggs
           Create eggs using crossover
           (Pc) with the best eggs in
           the nest and choose the
           best
          else
          if eggs = cuckoo_eggs,
          host_eggs
           Create eggs with mutation
           operator (Pm_2) for any
           cuckoo eggs in the nest and
           choose the best one among
           them
          else
           Create eggs randomly
          end if
      Construct the Pareto optimal
      Initialize the cuckoo (assume i)
      randomly by Lévy flights
      Evaluate the fitness, F_i
       Choose an egg with the worst
       solution in the nest (assume, j)
       if (F_i > F_j)
          Replace j by the new
          solution i
       end if
     Abandon or kill the p_a of worst
     nests
     Update the Pareto archive
     Build new nests at new locations
     using Lévy flights
     Evaluate the new population
   Repeat Until
   Obtain best solutions
End
```

Algorithm 3 Pareto Optimal Archive

```
INPUT: previousPopulation, list with
eggs as the genes from gene ranking
OUTPUT: newPopulation, list with
updated position of genes
Begin
  N = sizeof(previousPopulation)
  Fitness, F(x_i) and Crowding distance,
  C(x_i) for N, host nests,
  x_i = (i = 1, 2, ..., N)
  previousPopulation towards
  newPopulation
  for i = 1: N do
   for j = 1: N do
    if (F(x_i) > F(x_j))
     Move gene i toward gene j by
     updating its' position in all the
     dimensions
    else if (F(x_i) = F(x_j))
     if (C(x_i) > C(x_j))
      Update the position of gene j
      towards gene i in all dimensions
     end if
    end if
   end for
  end for
  Return newPopulation
End
```

The pseudo-code to construct the proposed Pareto optimal in cuckoo search is elaborated in Algorithm 3. Pareto optimal archived the concept of non-dominated sorting by comparing two solutions with multi-objectives and identify the one with a better process. Therefore, it is much possible to identify a set of solutions using a non-dominated set of solutions. In this approach, each solution is compared with every other solution in the population to check whether it gets dominated by other solutions. The developed non-dominated sorting and crowding distance in the proposed MOCS-EO is to identify the Pareto optimal solution aligned multi-objective constraint.

Another important component in the development of MOCS-EO is the evolutionary operators. In this case, the operators that will be utilized are double mutation and single crossover operators. The idea behind the use of these operators is to select the best eggs (eggs represent as genes). Usually, the eggs are created with random solutions. Thus, the following are the functionalities of operators in this research:

(a) First mutation operator: To generate a new solution by helping the cuckoo to imitate the host bird's eggs with uniform value, 0.01 [38].
(b) Crossover operator: Used to create cuckoo eggs in the nest and choose the best one among them.

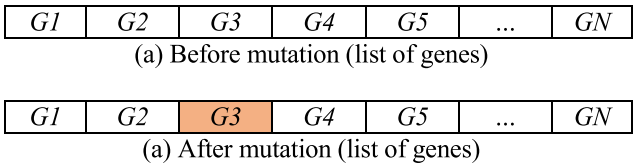


FIGURE 2. Before and after mutation.

(c) Second mutation operator: Used to create cuckoo eggs using crossover with mutation operator and choose the best egg among them using dynamic values.

To begin with, the first mutation operator is introduced in the initialization stage. This is to pre-search for the optimal genes to be used in the next stage of the process. It is an extra credit for the cancerous genes (highlighted *G3*) to be identified after mutation as shown in Figure 2.

Next crossover operator, the probability of crossover is to give freedom for the genes to be selected or not during the extreme selection process. It explains more about the reproduction process whereby makes clones of best genes but does create new genes with its formulation. While the second mutation operator defined the possible changes between the genes measurement of upper bound and lower bound which affect the performance during Pareto optimal. The upcoming section discusses the output from the implementation of the developed gene selection algorithm.

V. EXPERIMENTAL DESIGN

This section discusses the experimental design to prove the proposed gene selection algorithm's credibility. Table 1 shows seven publicly available microarray data sets used in this study [39]–[44]. However, assume all the data sets had undergone normalization under the range one to zero, gene clustering, and ranking.

The comparison between the MOCS-EO algorithm with the MOCS and CS algorithm will be presented in this section. Due to the limited number of samples present in the data, 10 cross-validation will be conducted for all data sets to determine the significant genes by the algorithms. Table 2 shows the control parameters for the CS algorithm used in upcoming experiments. The parameters involved in CS are the iterations, Lévy exponent discovery rate of alien solutions, and nest sizes.

Each experiment is carried out using a different combination of CS parameters. Usually, the parameters of cuckoo search are kept constant in all research, this might lead to decreasing the efficiency of the CS algorithm [45]. Smaller iterations such as 10, 20, and 30 with the combination of smaller nest sizes of 5, 10, 20, and 30 were focused and tested. These parameters were self-tuned to obtain the significant genes from the microarray data set.

Furthermore, focusing on the parameter of EO operators, the value of the initial mutation operator (Pm) is 0.01. The value of crossover (Pc) and mutation (Pm_2) operators are fixed as 0 and 1. The performances are evaluated based on accuracy, sensitivity, and F-measure.

TABLE 1. Number of genes, samples and classes in cancer microarray data.

Datasets	Genes	Samples	Class	Descriptions	References
Ovary cancer	15154	253	2	91 as normal and 162 as ovarian cancer	[39]
Lung cancer	12600	203	5	5 types of lung cancers	[40]
SRBCT	2308	83	4	4 types of childhood cancer tumors	[41]
CNS	7129	60	2	39 class 0 (survivors) and 21 class 1 (failures)	[42]
DLBCL	2647	77	2	58 as cancer and 19 as normal	[43]
Prostate cancer	2135	102	2	52 as prostate tumor and 50 as normal	[43]
Leukemia	12582	72	3	24 ALL, 20 AML and 28 MLL, 3 leukemia types	[44]

Note: SRBCT - Small red blue cell tumour; CNS - Central nervous system; DLBCL - Diffuse large B cell lymphoma

TABLE 2. Parameters for cuckoo search.

Parameters	Values
Nest sizes	5, 10, 20, 30
Discovery rate of eggs	0.25
Step size	1
Levy distribution coefficient	1.5
Iterations	10, 20, 30

The formula used as follows [46], [47]:

$$\text{Accuracy} = (TP + TN)/(TP + TN + FP + FN) \quad (5)$$

$$\text{Precision} = (TN)/(FP + TN) \quad (6)$$

$$\text{Recall} = (TP)/(TP + FN) \quad (7)$$

$$\text{F-measure} = (2 * Precision * Recall)/(Precision + Recall) \quad (8)$$

where, TP = true positive, TN = true negative, FP = false positive, and FN = false negative.

The classifiers used in this research is a deep neural network (DNN). The network will be initialized by ReLu and the chosen optimizer for DNN is SGD (learning rate = 0.01, momentum = 0.9). To prove the significance of the obtained results from the classifier, Wilcoxon signed rank-test will be performed. Wilcoxon test is a test used to compare two related samples, matched samples, or frequent measurements. This is a powerful test that can be used to identify the differentially expressed genes. The rule for this test is "*if the p-value is less than 0.05, then the result produced is significant*".

VI. RESULTS AND FINDINGS

In this section, this research applied MOCS-EO for gene selection on seven microarray data sets. The performances of the algorithm will be compared in terms of accuracy, the number of selected genes, and F-measure. Nevertheless, the DNN classifier is used to evaluate the number of selected genes for CS, MOCS, and MOCS-EO with different parameter settings as mentioned in Table 2. There are three sets of data evaluated before any implementation of the gene selection approach. In addition to it, to verify the obtained results were significant for the proposed gene selection algorithm, a statistical test has been conducted. The intention to use small iterations is to summarize the variation of results and strength of CS, MOCS, and MOCS-EO for cancer microarray data sets.

A. RESULTS FROM CS, MOCS, MOCS-EO ALGORITHM - 10 ITERATIONS

Table 3 summarizes the accuracy rate and F-measure obtained using different combinations of nest sizes and 10 iterations. DNN classifier used ten-fold cross-validations to evaluate the selected genes from CS, MOCS, and MOCS-EO algorithm. The best results for each data set are shown in bold. In Table 3, it can be seen that using 10 iterations for MOCS-EO algorithm, 89.29% of the data sets with different nest sizes able to produce the highest accuracy rates and F-measures rates compared to CS and MOCS algorithms (25 out of 28 showed high rates). Through oary cancer data, the highest performance measure is 97.5% for both accuracy and F-measure rates.

For lung cancer data with 5 nest sizes, the highest performance measures are achieved by the CS algorithm. However, compared between the nest sizes for 10 iterations in Lung cancer data is 93.7% (MOCS-EO algorithm). While for SRBCT data, the highest performance measure is acquired with the parameter settings of 10 nest sizes with the performance measures, 100% (MOCS-EO algorithm). Similar to other cancer microarray data sets, CNS, DLBCL, prostate cancer, and leukemia achieved the highest performance measures under 10 iterations using the proposed algorithm, MOCS-EO.

The highest performance measures achieved for accuracy and F-measure respectively are CNS, 73.7% and 73.3% (10 nest sizes), DLBCL is 94.5% (5 nest sizes), prostate cancer is 92.2% (20 nest sizes) and leukemia is 98.6% (5 nest sizes). It can be summarized from Table 3 using the parameter setting of 10 iterations with 5, 10, 20, 30 nest sizes, the data plays an important role in influencing the number of genes selected and performance results.

B. RESULTS FROM CS, MOCS, MOCS-EO ALGORITHM - 20 ITERATIONS

Table 4 summarizes the accuracy rate and F-measure obtained using a different combination of nest sizes and 20 iterations as the parameter settings for CS, MOCS, and MOCS-EO algorithms. From Table 4, approximately 85.71% of all

TABLE 3. Number of selected genes and performance measure results for 10 iterations.

Datasets	Nest sizes	Number of selected genes			Accuracy rates			F-measure		
		CS	MOCS	MOCS-EO	CS	MOCS	MOCS-EO	CS	MOCS	MOCS-EO
Ovary cancer	5	31	4219	1801	94.5	88.1	**97.1**	94.5	88.0	**97.1**
	10	31	3008	1951	94.5	87.7	**95.6**	94.5	87.0	**95.6**
	20	31	2803	1800	94.5	87.7	**96.5**	94.5	87.0	**96.5**
	30	31	3267	1308	94.5	88.5	**97.5**	94.5	87.9	**97.5**
Lung cancer	5	950	931	34	**92.6**	92.1	82.5	**92.8**	92.1	82.5
	10	186	130	2860	88.7	87.7	**91.9**	89.2	88.2	**92.0**
	20	130	116	3023	87.7	87.7	**93.7**	88.2	88.1	**93.9**
	30	140	144	2771	87.7	86.2	**91.9**	88.2	87.0	**92.0**
SRBCT	5	83	83	34	97.6	97.6	**98.4**	97.6	97.6	**98.4**
	10	83	83	216	97.6	97.6	**100.0**	97.6	97.6	**100.0**
	20	83	83	59	**97.6**	**97.6**	97.1	**97.6**	**97.6**	97.1
	30	83	83	13	**97.6**	**97.6**	83.0	**97.6**	**97.6**	82.9
CNS	5	1482	1371	955	48.3	51.7	**70.3**	49.5	52.7	**70.1**
	10	1716	1523	581	55.0	51.7	**73.7**	56.0	52.7	**73.3**
	20	1067	1551	150	57.9	55.0	**61.7**	57.5	56.0	**62.5**
	30	1191	1181	359	56.7	55.0	**73.7**	57.5	56.0	**73.3**
DLBCL	5	1094	610	263	83.1	80.5	**94.5**	83.9	81.7	**94.5**
	10	225	533	16	87.0	85.7	**88.5**	87.5	86.4	**88.5**
	20	149	679	132	84.4	83.1	**92.9**	85.3	84.0	**92.9**
	30	18	633	152	62.3	88.3	93.8	64.9	88.8	93.8
Prostate cancer	5	683	557	261	78.4	86.3	**90.2**	78.4	86.3	**90.2**
	10	818	553	108	90.2	88.2	**89.2**	90.2	88.2	**89.2**
	20	936	354	19	85.3	91.2	**92.2**	85.3	91.2	**92.2**
	30	443	535	102	87.3	84.3	**89.2**	87.3	84.3	**89.2**
Leukemia	5	189	5624	98	81.9	93.1	**98.6**	81.8	93.0	**98.6**
	10	189	5339	35	81.9	93.1	**95.8**	81.8	93.0	**95.8**
	20	189	5111	52	81.9	93.1	**97.2**	81.8	93.1	**97.2**
	30	189	5279	47	81.9	93.1	**94.5**	81.8	93.1	**94.5**

data shows that MOCS-EO algorithm able to produce the highest performance measures (24 out of 28 data with high rates). Nevertheless, the obtained highest accuracy rate and F-measure for SRBCT is 100% from 10 iteration and 10 nest sizes based on Table 4. Looking within the results of Table 4 for each cancer microarray data, ovary cancer data of 20 iterations achieved 97.8% of accuracy rates, and F-measures with 20 nest sizes via MOCS-EO algorithm.

While for lung cancer data, 95.2% is the highest performance measure rates achieved through 20 nest sizes and 20 iterations. For the CNS data set, MOCS-EO achieved 67.1% as the highest rate with 5 nest sizes compared to other data using different nest sizes. Similar to other microarray data sets, the highest performance measure rates achieved are from the MOCS-EO algorithm. Following nest size produced the highest performance measure rates are, 94.7% for DLBCL (5 nest sizes), 92.2% prostate cancer (10 nest sizes), and 98.6% for all nest sizes in leukemia.

C. RESULTS FROM CS, MOCS, MOCS-EO ALGORITHM - 30 ITERATIONS

Table 5 summarizes the accuracy rate and F-measure obtained using different combinations of nest sizes and 30 iterations. Similar to 20 iterations, approximately 85.71% of all data shows that the MOCS-EO algorithm able to produce the highest performance measures except for the SRBCT data set which is 97.6% with 83 selected genes. In Table 5, the MOCS-EO algorithm shows promising accuracy rates and F-measure for ovary cancer which is 97.5% (30 nest sizes). For lung cancer, MOCS-EO algorithm with 10 nest sizes produced the highest performance measure rates with 93.7% and 4014 selected genes.

Looking at the CNS data set, the highest performance measures were obtained by the MOCS-EO algorithm under 20 nest sizes with 76.3%. With the DLBCL data set, the highest accuracy and F-measure rates obtained are 95.6% with 5 nest sizes. While for prostate cancer, 93.3% has been achieved as the highest accuracy and F-measures rates with 30 iterations and 10 nest sizes. Besides, for leukemia, 10, 20, and 30 nest sizes showed similar and highest accuracy and F-measure rates which were 95.8%.

D. DISCUSSION

Summarizing the results obtained from the parameter setting of 10, 20, and 30 iterations and 5, 10, 20, and 30 nest sizes for the listed microarray data sets using DNN showcased

TABLE 4. Number of selected genes and performance measure results for 20 iterations.

Datasets	Nest sizes	Number of selected genes			Accuracy rates			F-measure		
		CS	MOCS	MOCS-EO	CS	MOCS	MOCS-EO	CS	MOCS	MOCS-EO
Ovary cancer	5	31	4565	1901	94.5	88.9	**97.1**	94.5	88.4	**97.1**
	10	31	3433	1674	94.5	88.1	**97.5**	94.5	87.5	**97.5**
	20	31	3257	1440	94.5	87.0	**97.8**	94.5	87.0	**97.8**
	30	31	3641	1400	94.5	88.9	**96.5**	94.5	88.4	**96.5**
Lung cancer	5	410	195	3160	86.7	89.7	**92.8**	87.3	90.0	**92.8**
	10	235	197	3215	90.1	88.2	**92.3**	90.5	88.5	**92.3**
	20	184	171	3282	89.2	86.2	**95.2**	89.6	87.1	**95.2**
	30	171	202	3071	89.7	89.2	**91.9**	90.0	89.6	**91.9**
SRBCT	5	83	83	37	**97.6**	**97.6**	88.0	**97.6**	**97.6**	87.9
	10	83	83	69	**97.6**	**97.6**	63.2	**97.6**	**97.6**	62.9
	20	83	83	113	**97.6**	**97.6**	85.2	**97.6**	**97.6**	85.1
	30	83	83	41	**97.6**	**97.6**	90.9	**97.6**	**97.6**	90.9
CNS	5	1828	1561	754	60.0	50.0	**67.1**	60.8	51.1	**66.5**
	10	1972	1698	362	56.6	53.3	**58.3**	56.3	53.3	**59.3**
	20	1351	1726	157	63.3	51.7	**64.5**	64.2	52.7	**63.9**
	30	1512	1383	83	56.7	55.0	**65.8**	57.1	55.9	**65.6**
DLBCL	5	1127	681	334	87.0	80.5	**96.3**	87.5	81.8	**96.3**
	10	338	579	79	83.1	83.1	**93.6**	84.0	84.1	**93.4**
	20	250	727	26	79.2	83.1	**84.1**	80.6	84.0	**84.0**
	30	181	667	165	83.1	84.4	**94.7**	83.1	85.3	**94.7**
Prostate cancer	5	741	602	64	78.4	85.3	**89.2**	78.4	85.3	**89.2**
	10	854	597	81	81.4	86.3	**92.2**	81.3	86.3	**92.2**
	20	946	435	135	78.4	89.2	**91.2**	78.2	89.2	**91.2**
	30	552	575	206	86.3	**88.2**	**88.2**	86.2	**88.2**	**88.2**
Leukemia	5	189	6214	98	81.9	93.1	**98.6**	81.8	93.1	**98.6**
	10	189	6051	97	81.9	94.4	**98.6**	81.8	94.4	**98.6**
	20	189	6159	98	81.9	95.8	**98.6**	81.8	95.8	**98.6**
	30	189	6227	98	81.9	94.4	**98.6**	81.8	94.4	**98.6**

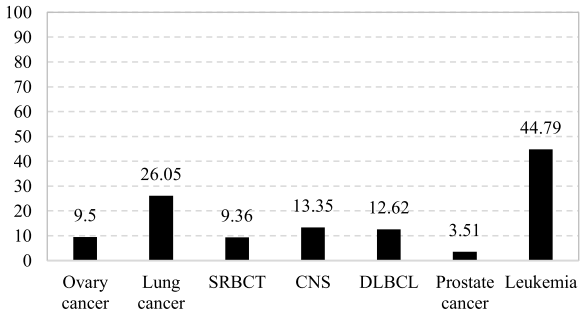


FIGURE 3. Ratio of selected genes.

that the parameters are very important and influence the performance measure results. Table 6 shows the highest performance measures obtained from 10, 20, and 30 iterations from the MOCS-EO algorithm. It can be concluded that out of seven data sets, three data sets able to produce the highest performance measures with 20 iterations. Nevertheless, every cancer microarray data sets require different nest sizes which might influence the number of selected genes that consist of informative genes.

Furthermore, Figure 3 shows the ratio of selected genes with the highest performance measures for all cancer microarray data sets using the proposed algorithm. The selected genes using the MOCS-EO algorithm are believed to be significant and can assist in decision-making using the genes. Whilst, the ratio of selected genes able to depict that the number of informative genes in the cancer microarray data is lesser than 50% of the genes. For example, as can be seen in Figure 3, for ovary cancer data, only 9.5% of genes were significant genes. Thus, 90.5% of genes are non-significant and cause noise in order to make good decision-making. Therefore, identifying the genes requires an algorithm that able to handle these data and select the best genes.

VII. ANALYSIS ON RESULTS

In this research, the MOCS-EO algorithm was used for gene selection with the parameter settings of 10, 20, and 30 iterations with 5, 10, 20, and 30 nest sizes. Each finding from the parameter settings results in different fitness values. The higher the fitness values due to its information, the higher chances of the genes being selected.

To look upon each data in Table 6, MOCS-EO showcased the highest accuracy rates compared to other algorithms. Discussing the MOCS-EO algorithm, the highest parameter settings for ovary cancer is 20 iterations and 20 nest sizes, the highest accuracy rates is 97.8% (selected 1440 genes). Next, the lung cancer data, the MOCS-EO algorithm has

TABLE 5. Number of selected genes and performance measure results for 30 iterations.

Datasets	Nest sizes	Number of selected genes			Accuracy rates			F-measure		
		CS	MOCS	MOCS-EO	CS	MOCS	MOCS-EO	CS	MOCS	MOCS-EO
Ovary cancer	5	31	4870	2432	94.5	89.7	**97.1**	94.5	89.2	**97.1**
	10	31	3828	2458	94.5	88.1	**96.9**	94.5	87.5	**96.9**
	20	31	3649	2224	94.5	87.7	**97.2**	94.5	87.0	**97.2**
	30	31	4041	2201	94.5	89.7	**97.5**	94.5	89.2	**97.5**
Lung cancer	5	416	236	3993	90.6	87.7	**92.0**	90.9	88.4	**92.0**
	10	276	232	4014	88.7	88.7	**93.7**	89.2	89.1	**93.7**
	20	206	190	4064	87.7	88.7	**92.6**	88.2	89.1	**92.6**
	30	203	180	3936	86.2	89.2	**91.1**	86.7	89.7	**91.3**
SRBCT	5	83	83	37	**97.6**	**97.6**	88.0	**97.6**	**97.6**	87.9
	10	83	83	69	**97.6**	**97.6**	63.2	**97.6**	**97.6**	62.9
	20	83	83	113	**97.6**	**97.6**	85.2	**97.6**	**97.6**	85.1
	30	83	83	41	**97.6**	**97.6**	90.9	**97.6**	**97.6**	90.9
CNS	5	2111	1741	1495	55.3	55.0	**63.3**	54.9	56.0	**63.9**
	10	2280	1855	1236	66.7	56.7	**71.1**	69.4	59.6	**70.6**
	20	1568	1871	952	55.0	60.0	**76.3**	56.0	60.7	**75.8**
	30	1708	1563	83	53.3	53.3	**65.8**	54.3	54.4	**65.6**
DLBCL	5	1137	725	122	85.7	84.4	**95.6**	86.5	85.3	**95.6**
	10	374	650	97	76.6	85.7	**91.3**	78.2	86.4	**91.4**
	20	287	797	281	85.7	85.7	**94.7**	86.5	89.7	**94.8**
	30	245	719	255	84.4	77.9	**94.7**	85.4	79.3	**94.7**
Prostate cancer	5	791	647	17	68.6	86.3	**88.2**	68.1	86.3	**88.2**
	10	880	633	75	88.2	90.2	**93.3**	88.2	90.2	**93.3**
	20	973	491	6	75.5	87.3	**92.2**	75.5	87.3	**92.2**
	30	586	604	12	76.5	87.3	**88.2**	76.5	87.3	**88.2**
Leukemia	5	189	6302	187	81.9	91.7	**93.1**	81.8	91.5	**93.1**
	10	189	6226	144	81.9	87.5	**95.8**	81.8	87.6	**95.8**
	20	189	6291	102	81.9	90.3	**95.8**	81.8	90.3	**95.8**
	30	189	6185	29	81.9	94.4	**95.8**	81.8	94.4	**95.8**

achieved the highest accuracy rates using 20 iterations, 20 nest sizes (3282 genes) attained 95.2%. While for SRBCT data, the parameter settings of the highest performance measure achieved are 10 iterations, 10 nest sizes with 216 selected genes attaining 100.0% accuracy rate. Furthermore, CNS data obtained the highest performance measure, 76.3% accuracy rates using the settings of 30 iterations, and 20 nest sizes (952 selected genes). For DLBCL data, the highest performance measure, 96.3% accuracy rate achieved with the parameter setting of 20 iterations and 5 nest sizes with 334 selected genes. Prostate cancer attained high performance which is 93.3%, accuracy rate using 30 iterations, and 10 nest sizes with 75 selected genes.

Finally, for the leukemia data set, MOCS-EO displayed a higher accuracy rate compared to other algorithms. Leukemia data achieved 98.6% accuracy rate with 10 iterations and 5 nest sizes with 5624 selected genes. Concluding Table 6, it can be seen that, 20 iterations and 20 nest sizes exhibit the highest performance measures for two cancer microarray data sets, while 30 iterations were shown best for two cancer microarray data sets which is CNS and prostate cancer with different selected nest sizes, 20 and 10 respectively.

Every microarray data attained high performance measures with different parameter settings. Therefore, it is evident

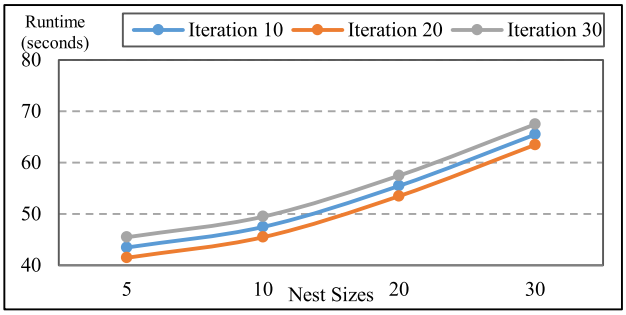
TABLE 6. Summary of highest performance measures.

Data	Acc. (%)	F-measure (%)	Iterations, nest sizes	Selected genes
Ovary cancer	97.8	97.8	20, 20	1440
Lung cancer	95.2	95.2	20, 20	3282
SRBCT	100.0	100.0	10, 30	216
CNS	76.3	75.8	30, 20	952
DLBCL	96.3	96.3	20, 5	334
Prostate cancer	93.3	93.3	30, 10	75
Leukemia	98.6	98.6	10, 5	5624

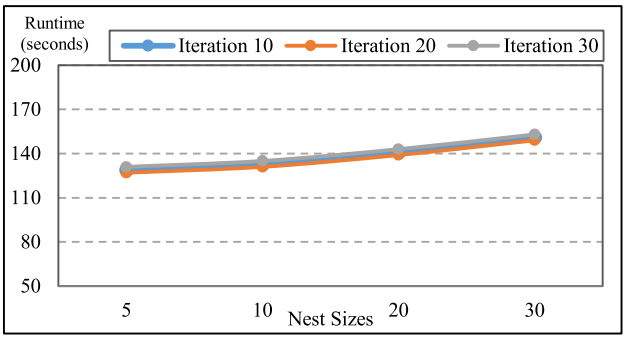
that every cancer microarray data consists of its parameter setting requirements to process the data rigorously to identify the genes that can contribute toward the cancer classes. Nevertheless, all cancer microarray data had reduced its dimensionalities and identified the informative genes for its classes through the developed MOCS-EO as gene selection algorithm.

VIII. COMPUTATIONAL COMPLEXITY

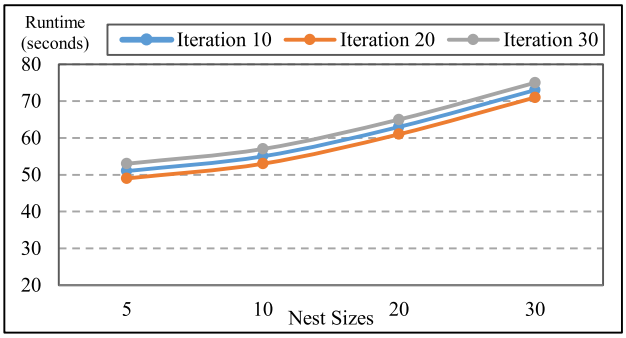
Based on the total run time for gene selection using the developed MOCS-EO algorithm for all cancer microarray



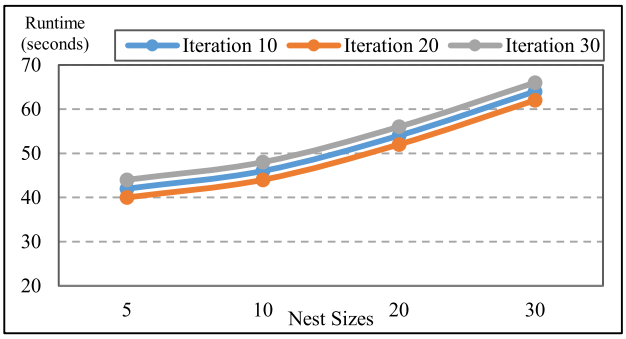
(a) Ovary cancer data



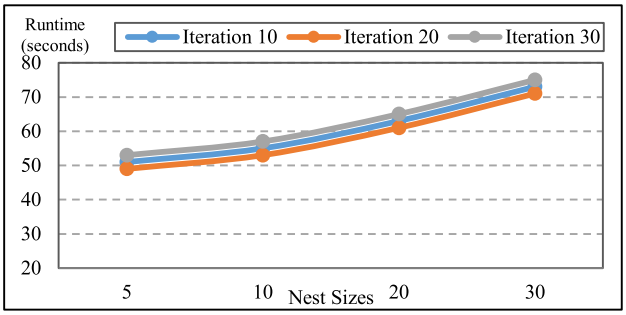
(b) Lung cancer data



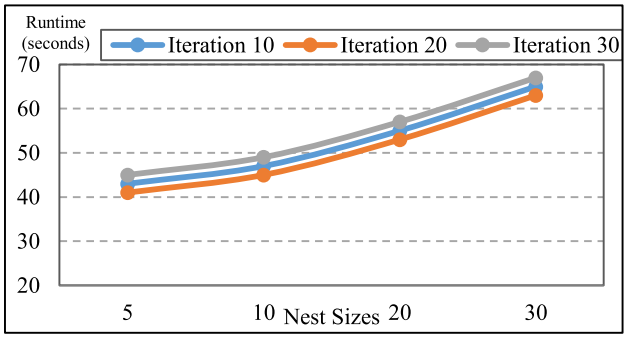
(c) SRBCT data



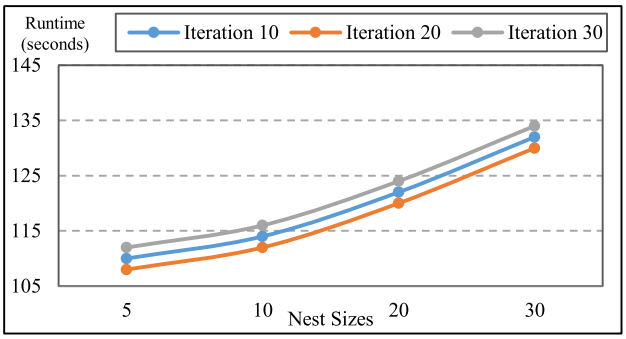
(d) CNS data



(e) DLBCL data



(f) Prostate cancer data



(g) Leukemia data

FIGURE 4. Linear fitting of run time versus number of iterations for all cancer microarray data sets.

data sets used in this research, the algorithm runs in through the fitted model by estimating the order of time. Following is the MOCS-EO algorithm expressed through big-O notation which able to express the growth rate of a function:

$$f(x) = p_1 x + p_2 x \oplus p_3 x \tag{9}$$

The time complexity represented by Equation (9) is $O(n)$. Therefore, the running time increases linearly corresponding to the size of data.

Figure 4 shows the running time (in seconds) with the number of iterations to predict the asymptotic behavior of the run time of each data set. It is clear that based on the fitted

TABLE 7. Summary of Wilcoxon verification test.

Data	Original Acc. (%)	p-values	Significant
Ovary cancer	64.0	0.018	*
Lung cancer	68.5	0.018	*
SRBCT	92.8	0.042	*
CNS	48.3	0.006	*
DLBCL	88.3	0.048	*
Prostate cancer	64.7	0.018	*
Leukemia	88.9	0.027	*

TABLE 8. Summary of t-test verification test.

Data	Original Acc. (%)	p-values	Significant
Ovary cancer	64.0	0.0160	*
Lung cancer	68.5	0.0040	*
SRBCT	92.8	0.0000	*
CNS	48.3	0.0032	*
DLBCL	88.3	0.1930	
Prostate cancer	64.7	0.0031	*
Leukemia	88.9	0.0650	

TABLE 9. Comparison of results with other research.

Research/ Methods	Data	Accuracy (%)	MOCS-EO Accuracy (%)
[49] IGIS + kNN	SRBCT	91.35	100.0
	CNS	57.83	76.30
	Ovary cancer	99.49	97.80
	Leukemia	83.91	98.60
	Lung cancer	91.69	95.20
[50] Elephant search+ DNN	SRBCT	83.14	100.0
	CNS	53.34	76.30
	Ovary cancer	99.21	97.80
	Lung cancer	94.10	95.20
	DLBCL	91.49	95.60
	Prostate	88.24	93.5
[50] Firefly search+ DNN	SRBCT	93.98	100.0
	CNS	56.67	76.30
	Ovary cancer	97.24	97.80
	Lung cancer	93.11	95.20
	DLBCL	89.36	95.60
	Prostate	87.26	93.5

model used to predict the asymptotic behavior of run time of each dataset the MOCS-EO algorithm on each iteration runs linearly from 5 until 30 nest sizes aligned with 10, 20, and 30 iterations. Nevertheless, 20 iterations show a better time complexity compared to other iterations of nest sizes. Performing time complexity analysis the MOCS-EO algorithm determined that the performance of the algorithm in real-time increases along with nest sizes.

IX. VERIFICATION AND VALIDATION OF RESULTS

The verification of final results is summarized in Table 7 and 8. The obtained significance using Wilcoxon and paired sample t-test for accuracy with the final results of all cancer microarray data to prove and verify the credibility of the developed MOCS-EO algorithm. The significance is indicated as ''*''. For both tests, p-values < 0.05 indicate the significance of the results (95% confidence of results). Based on Table 7 and 8, the original accuracy results compared to the selected groups of experiments in Table 6. All results using the Wilcoxon test shows the algorithm is significant with the identified genes. Here, the Wilcoxon test is a test used to compare two related samples, matched samples, or frequent measurements. This test is a powerful test that can be used with microarray data to identify the differentially expressed genes.

While for t-test, all results using MOCS-EO were significant except DLBCL and leukemia data. This is because both data were repeated using a much larger sample size which might decrease the width of the confidence interval for both data sets as supported by [48]. Therefore, the obtained results (selected genes) for leukemia and DLBCL has less impact on the data due to the properties of its data sets. Here, the t-test is conducted to verify the relationship between the selected genes using the developed algorithm.

To validate the results obtained from the proposed MOCS-EO algorithm, the microarray data used for this research is compared with other recent research to showcase the credibility of the developed algorithm for gene selection. As shown in Table 9, MOCS-EO exhibits better accuracy rates compared to other methods such as improved interaction information guided incremental selection (IGIS), or meta-heuristic methods namely firefly search and elephant search using similar cancer microarray data sets. However, except for ovary cancer data, the IGIS and elephant search showcased better gene selection. This might be due to lower parameter settings used in cuckoo search.

To improvise the performance measures, the parameter settings of MOCS-EO can be modified into larger settings. This might be due to the larger sample size in ovary cancer data that decrease the MOCS-EO capacity in searching the best genes in the data. Therefore, higher iteration and higher nest sizes for MOCS-EO might able to solve the problem in the algorithm. From a larger perspective, it can be seen that there are clear differences of values between the accuracy rates obtained from other researchers and developed MOCS-EO algorithm. This is because MOCS-EO has the advantage in the algorithm such as multi-objective optimization, double mutation, and crossover operators that able to search best genes out of the large amount of data set.

X. CONCLUSION

Classification is a data mining task which is an important task for gene selection. The main reason to propose a gene selection algorithm is to select significant genes from the large

gene space that consists of noisy gene expression data. From these experiments can be concluded that, in a gene selection procedure, it is very important to build a good classifier model to further the biological investigation. These experiments confirmed the hypothesis that the proposed algorithm can select the important genes compared to other genes based in a class, however, the fallback is the classifier. The classifier can predict only the correlations within the selected gene sets. With the DNN classifier, the performance measure rates of the proposed algorithm able to justify as DNN to analyze the correlations between genes. For real-world biological and clinical applications, these simulations conducted in this research able to assist in identifying the genes that contribute toward cancer diseases.

The limitation in MOCS-EO as a gene selection algorithm is the algorithm bounded to self-tune the parameter, therefore, parameter tuning of CS can be enhanced. Nevertheless, for the contribution using the MOCS-EO algorithm as a feature selection method, the use of a multi-objective approach aims to minimize the number of genes and maximize the relevance of the selected genes and the cancer classes. While EO such as mutation and crossover enhances the responsibility to choose the best out of best genes. This developed algorithm has successfully chosen a smaller number of genes for decision-making and better classification performance than using all genes with different smaller parameter settings for all seven cancer microarray data sets.

REFERENCES

[1] I. Hameed, S. R. Masoodi, P. A. Malik, S. A. Mir, K. Ghazanfar, and B. A. Ganai, "Genetic variations in key inflammatory cytokines exacerbates the risk of diabetic nephropathy by influencing the gene expression," *Gene*, vol. 661, pp. 51–59, Jun. 2018.

[2] J. Tang, A. Salem, and L. Huan, "Feature selection for classification: A review," *Data Classification, Algorithms Appl.*, vol. 51, pp. 37–64, Oct. 2014.

[3] A. P. Angulo, "Gene selection for microarray cancer data classification by a novel rule-based algorithm," *Information*, vol. 9, no. 1, pp. 1–15, 2018.

[4] B. Wojtas, A. Pfeifer, and M. Oczko-Wo, "Gene expression (mRNA) markers for differentiating between malignant and beningn follicular thyroid tumor," *Int. J. Mol. Sci.*, vol. 18, p. 1184, Dec. 2017.

[5] Babu, M. M, An introduction to microarray data analysis. in computational genomics: Theory and application,' *Horizon Bioscience*, vol. 15, pp. 225–249, Oct. 2004.

[6] B. Schlegel and B. Sick, "Design and optimization of an autonomous feature selection pipeline for high dimensional, heterogeneous feature spaces," in *Proc. IEEE Symp. Ser. Comput. Intell. (SSCI)*, Dec. 2016, pp. 1–9.

[7] D. Oreski, S. Oreski, and B. Klicek, "Effects of dataset characteristics on the performance of feature selection techniques," *Appl. Soft Comput.*, vol. 52, pp. 109–119, Mar. 2017.

[8] N. D. Cilia, C. D. Stefano, F. Fontanella, S. Ralmonodo, and A. S. Freca, "An experimental comparison of feature-selection and classification methods for microarray datasets," *Inf.*, pp. 1–13, 2019.

[9] G. V. Trunk, "A problem of dimensionality: A simple example," *IEEE Trans. Pattern Anal. Mach. Intell.*, vols. PAMI–1, no. 3, pp. 306–307, Jul. 1979.

[10] P. Bermejo, J. A. Gámez, and J. M. Puerta, "Adapting the CMIM algorithm for multilabel feature selection. A comparison with existing methods," *Expert Syst.*, vol. 35, no. 1, p. 2230, 2018.

[11] D. R. Prasad and P. V. Naganjaneyulu, "Metaheuristics techniques for cluster selection in WSN," *Int. Conf. Algorithms, Methodol., Models Appl. Emerg. Technol.*, vol. 1, pp. 1–6, Dec. 2017.

[12] S. S. Shreem, S. Abdullah, and M. Z. A. Nazri, "Hybrid feature selection algorithm using symmetrical uncertainty and a harmony search algorithm," *Int. J. Syst. Sci.*, vol. 47, no. 6, pp. 1312–1329, Apr. 2016.

[13] A. Nagpal and V. Singh, "A feature selection algorithm based on qualitative mutual information for cancer microarray data," *Procedia Comput. Sci.*, vol. 132, pp. 244–252, 2018.

[14] M. N. Sudha and S. Selvarajan, "Feature selection based on enhanced cuckoo search for breast cancer classification in mammogram image," *Circuits Syst.*, vol. 07, no. 04, pp. 327–338, 2016.

[15] S. B. Sakri, N. B. Abdul Rashid, and Z. Muhammad Zain, "Particle swarm optimization feature selection for breast cancer recurrence prediction," *IEEE Access*, vol. 6, pp. 29637–29647, 2018.

[16] S. S. Hameed, O. O. Petinrin, A. Osman, F. S. Hashi, and F. S. Filter-Wrapper, "Combination and embedded feature selection for gene expression data," *Int. J. Advance Soft Compu. Appl.*, vol. 10, no. 1, p. 5, 2018.

[17] S. N. Win, C. N. Siew, H. K. Kyaw, and W. L. Shir, "Particle swarm Feature Selection for Micrarrau Leukemia Clcassificatkon," *Prog. Engergy Environ., Penerbit Akademia Baru*, vol. 2, pp. 1–8, May 2017.

[18] Z. B. Ozger, B. Bolat, and B. Diri, "A Probabilistic Multi-Objective Artificial Bee Colony Algorithm for Gene Selection," *J. Universal Comput. Sci.*, vol. 25, no. 4, vol. 25, no. 4, pp. 418–443. 2019.

[19] S. S. Jagadeesh and R. Sugumar, "A Comparative Study on Artificial Bee Colony with Modified ABC Algorithm," *Eur. J. Appl. Science.*, vol. 9, no. 5, pp. 243–248, 2017.

[20] A. Kaveh and T. Bakhshpoori, "An efficient multi-objective cuckoo search algorithm for design optimization," *Adv. Comput. Design*, vol. 1, no. 1, pp. 87–103, Jan. 2016.

[21] Y. Zhang, S. Cheng, Y. Shi, D. W. Gong, and X. Zhao, "Cost-sensitive feature selection using two-archive multi-objective artificial bee colony algorithm," *Expert Syst. Appl.*, vol. 137, pp. 46–58, Dec. 2019.

[22] Y. Zhang, "Binary differential evolution with self-learning for multi-objective feature selection," *Inf. Sci.*, vol. 507, pp. 67–85, Dec. 2020.

[23] M. Feizi-Derakshi and M. Ghaemi, "Classifying different feature selection based on the search strategies," in *Proc. Int. Conf. Mach. Learn., Electr. Mech. Eng.*, pp. 17–21, 2014.

[24] S. Sasikala, S. A Balamurugan and S. Geetha, "A novel feature selection technique for improved survivability diagnosis of breast cancer," *Procedia Comput. Sci.*, vol. 50, pp. 16–23, Dec. 2015.

[25] Yahya, "Feature selection for high dimensional data: An evolutionary filter approach," *J. Comput. Sci.*, vol. 7, no. 5, pp. 800–820, May 2011.

[26] M. Fatimaezzahra, S. Mohamed, and E. Abdelaziz, "A combined cuckoo search algorithm and genetic algorithm for parameter optimization in computer vision," *Int. J. Appl. Eng. Res.*, vol. 51, pp. 12940–12954, Dec. 2017.

[27] X. S. Yang, *Nature-Inspired Metaheuristic Algorithms*. Beckington, U.K.: Luniver Press, 2010.

[28] M. Mareli and B. Twala, "An adaptive cuckoo search algorithm for optimisation," *Appl. Comput. Informat.*, vol. 14, no. 2, pp. 107–115, Jul. 2018.

[29] N. J. Sato, K. Tokue, R. A. Noske, O. K. Mikami, and K. Ueda, "Evicting cuckoo nestlings from the nest: A new anti-parasitism behaviour," *Biol. Lett.*, vol. 6, no. 1, pp. 67–69, Feb. 2010.

[30] I. Khoja, T. Ladhari, F. M'sahli, and A. Sakly, "Cuckoo search approach for parameter identification of an activated sludge process," *Comput. Intell. Neurosci.*, vol. 2018, pp. 1–9, Dec. 2018.

[31] S. Pare, A. K. Bhandari, A. Kumar, and G. K. Singh, "A new technique for multilevel color image thresholding based on modified fuzzy entropy and Lévy flight firefly algorithm," *Comput. Electr. Eng.*, vol. 70, pp. 476–495, Aug. 2018.

[32] X. H. Yan and F. Z. Y. L. He Chen, "A novel hardware/software partitioning method based on position disturbed particle swarm optimization with invasive weed optimization," *J. Comput. Sci. Technol.*, vol. 32, no. 2, pp. 35–340, 2017.

[33] M. Thums, J. Fernández-Gracia, A. M. M. Sequeira, V. M. Eguíluz, C. M. Duarte, and M. G. Meekan, "How big data fast tracked human mobility research and the lessons for animal movement ecology," *Frontiers Mar. Sci.*, vol. 5, pp. 1–12, Feb. 2018.

[34] A. Harkat, R. Benzid, and L. Saidi, "Features extraction and classification of ECG beats using CWT combined to RBF neural network optimized by cuckoo search via levy flight," in *Proc. 4th Int. Conf. Electr. Eng. (ICEE)*, Dec. 2015, pp. 1–4.

[35] N. A. Husaini, R. Ghazali, and I. T. R. Yanto, "Enhancing modified cuckoo search algorithm by using MCMC random walk," in *Proc. 2nd Int. Conf. Sci. Inf. Technol. (ICSITech)*, Oct. 2016, pp. 1–6.

[36] G. Wang, "A comparative study of cuckoo algorithm and ant colony algorithm in optimal path problems," *MATEC Web Conf.*, vol. 232, pp. 1–6, Dec. 2018.
[37] B. Rengeswaran, N. Mathaiyan, and P. Kandasampy, "Cuckoo search with mutation for biclustering of microarray gene expression data," *The Int. Arab J. Inf. Technol.*, vol. 14, no. 3, pp. 300–306, 2017.
[38] V. Selvi, "Comparative analysis of swarm intelligence techniques," *Global J. Eng. Sci. Res. Manage.*, vol. 2, no. 7, pp. 9–19, 2015.
[39] Z. Zhu, Y.-S. Ong, and M. Dash, "Markov blanket-embedded genetic algorithm for gene selection," *Pattern Recognit.*, vol. 40, no. 11, pp. 3236–3248, Nov. 2007.
[40] J. Khan, J. S. Wei, M. Ringer, L. H. Saal, M. Ladanyi, and G. Wetermann, "Classification and diagnosyic prediction of cancers using gene expression profiling and artificial neural network," *Nat. Med.*, vol. 7, no. 6, pp. 673–679.
[41] S. Pomeroy, "Prediction of central nervous system embryonal tumor outcome based on gene expression," *Nature*, vol. 415, pp. 436–442, Dec. 2010.
[42] D. Singh, P. G. Febbo, K. Ross, D. G. Jackson, J. Manola, C. Ladd, P. Tamayo, A. A. Renshaw, A. V. D'Amico, J. P. Richie, E. S. Lander, M. Loda, P. W. Kantoff, T. R. Golub, and W. R. Sellers, "Gene expression correlates of clinical prostate cancer behavior," *Cancer Cell*, vol. 1, no. 2, pp. 203–209, Mar. 2002.
[43] S. A. Armstrong, J. E. Staunton, L. B. Silverman, R. Pieters, M. L. den Boer, M. D. Minden, S. E. Sallan, E. S. Lander, T. R. Golub, and S. J. Korsmeyer, "MLL translocations specify a distinct gene expression profile that distinguishes a unique leukemia," *Nature Genet.*, vol. 30, no. 1, pp. 41–47, Jan. 2002.
[44] N. Scherbakov, S. von Haehling, S. D. Anker, U. Dirnagl, and W. Doehner, "Stroke induced sarcopenia: Muscle wasting and disability after stroke," *Int. J. Cardiol.*, vol. 170, no. 2, pp. 89–94, Dec. 2013.
[45] E. F. Petricoin, A. M. Ardekani, B. A. Hiltt, P. J. Levine, V. A. Fusaro, and S. M. Steinberg, "Use of proteomic patterns n serum to identify ovarian cancer," *Lancet*, vol. 359, pp. 572–577.
[46] N. Scherbakov, S. von Haehling, S. D. Anker, U. Dirnagl, and W. Doehner, "Stroke induced sarcopenia: Muscle wasting and disability after stroke," *Int. J. Cardiol.*, vol. 170, no. 2, pp. 89–94, Dec. 2013.
[47] E. Kouchaki, R. D. Kakhaki, O. R. Tamtaji, E. Dadgostar, M. Behnam, H. Nikoueinejad, and H. Akbari, "Increased serum levels of TNF-? and decreased serum levels of IL-27 in patients with Parkinson disease and their correlation with disease severity," *Clin. Neurol. Neurosurg.*, vol. 166, pp. 76–79, 2018.
[48] S. Aarts, B. Winkens, and M. van Den Akker, "The insignificance of statistical significance," *Eur. J. Gen. Pract.*, vol. 18, no. 1, pp. 50–52, Mar. 2012.
[49] S. Nakariyakul, "A hybrid gene selection algorithm based on interaction information for microarray-based cancer classification," *PLoS ONE*, vol. 4, no. 2, pp. 1–17, 2019.
[50] M. Panda, "Elephant search optimization combined with deep neural network for microarray data analysis," *J. King Saud Univ.—Comput. Inf. Sci.*, vol. 14, pp. 1–12, Dec. 2017.

MOHD SHAHIZAN OTHMAN received the B.Sc. degree in computer science with a major in information systems from the Universiti Teknologi Malaysia (UTM), Malaysia, in 1998, and the M.Sc. degree in information technology and the Ph.D. degree in web information extraction, information retrieval and machine learning from the Universiti Kebangsaan Malaysia (UKM), Malaysia. He is currently an Associate Professor with the Faculty of Engineering, School of Computing, UTM. His research interests include information extraction and information retrieval on the web, web data mining, content management, machine learning, social learning, e-learning, business intelligence, and geographic information system (GIS).

SHAMINI RAJA KUMARAN received the B.I.T. degree in information technology major in software engineering from the Universiti Utara Malaysia (UUM), Malaysia, in 2015, and the Master of Philosophy degree in computer science from the Universiti Teknologi Malaysia (UTM), in 2017, where she is currently pursuing the Doctor of Philosophy degree. Her research interests include machine learning, optimization, big data, business intelligence, and data mining.

LIZAWATI MI YUSUF received the B.Sc. degree in computer science with a major in industrial computing from the Universiti Teknologi Malaysia (UTM), Malaysia, in 2000, and the M.Sc. degree in information technology from the Universiti Kebangsaan Malaysia (UKM), Malaysia. She is currently a Lecturer with the Faculty of Engineering, School of Computing, UTM. Her research interests include optimization, web information extraction and retrieval, web data mining, machine learning, social learning, business intelligence, high-performance computing, and numerical analysis.