Princess Sumaya University for Technology
Since 1991 تأسست
PSUT
جامعة الأميرة سمية للتكنولوجيا

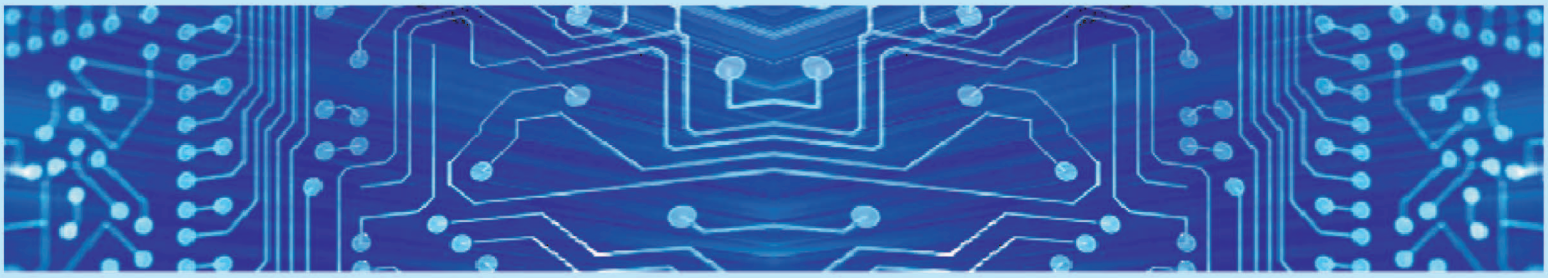
صندوق دعم البحث العلمي
Scientific Research Support Fund

Jordanian Journal of Computers and Information Technology

December 2018 VOLUME 04 NUMBER 03 ISSN 2415 - 1076 (Online)
ISSN 2413 - 9351 (Print)

JJCIT

www.jjcit.org jjcit@psut.edu.jo



An International Peer-Reviewed Scientific Journal
Financed by the Scientific Research Support Fund

Jordanian Journal of Computers and Information Technology (JJCIT)

The Jordanian Journal of Computers and Information Technology (JJCIT) is an international journal that publishes original, high-quality and cutting edge research papers on all aspects and technologies in ICT fields.

JJCIT is hosted by Princess Sumaya University for Technology (PSUT) and supported by the Scientific Research Support Fund in Jordan. Researchers have the right to read, print, distribute, search, download, copy or link to the full text of articles. JJCIT permits reproduction as long as the source is acknowledged.

Aims and Scope

The JJCIT aims to publish the most current developments in the form of original articles as well as review articles in all areas of Telecommunications, Computer Engineering and Information Technology and make them available to researchers worldwide. The JJCIT focuses on topics including, but not limited to: Computer Engineering & Communication Networks, Computer Science & Information Systems and Information Technology and Applications.

Indexing

JJCIT is indexed in:

DOAJ DIRECTORY OF OPEN ACCESS JOURNALS

crossref Text and Data Mining

OCLC WorldCat

BASE

Scilit

Google scholar

Editorial Board Support Team

JJCIT Address

Website: www.jjcit.org
Email: jjcit@psut.edu.jo
Address: Princess Sumaya University for Technology, Khalil Saket Street, Al-Jubaiha
B.O. Box: 1438 Amman 11941 Jordan
Telephone: +962-6-5359949
Fax: +962-6-7295534

EDITORIAL BOARD

INTERNATIONAL ADVISORY BOARD

"Opinions or views expressed in papers published in this journal are those of the author(s) and do not necessarily reflect those of the Editorial Board, the host university or the policy of the Scientific Research Support Fund".

"ما ورد في هذه المجلة يعبر عن آراء الباحثين ولا يعكس بالضرورة آراء هيئة التحرير أو الجامعة أو سياسة صندوق دعم البحث العلمي".

DISTRIBUTED GREY WOLF OPTIMIZER FOR NUMERICAL OPTIMIZATION PROBLEMS

Bilal H. Abed-alguni and Malek Barhoush

(Received: 29-Jul.-2018, Revised: 10-Sep.-2018, Accepted: 13-Sep.-2018)

ABSTRACT

The Grey Wolf Optimizer (GWO) algorithm is an interesting swarm-based optimization algorithm for global optimization. It was inspired by the hunting strategy and leadership hierarchy of grey wolves. The GWO algorithm has been successfully tailored to solve various continuous and discrete optimization problems. However, the main drawback of GWO is that it may converge to sub-optimal solutions in early stages of its simulation process due to the loss of diversity in its population. This paper introduces a distributed variation of GWO(DGWO) that attempts to enhance the diversity of GWO by organizing its population into small independent groups (islands) based on a well-known distributed model called the island model. DGWO applies the original GWO to each island and then allows selected solutions to be exchanged among the islands based on the random ring topology and the best-worst migration policy. The island model in DGWO provides a better environment for unfit candidate solutions in each island to evolve into better solutions, which increases the likelihood of finding global optimal solutions. Another interesting feature about DGWO is that it can run in parallel devices, which means that its computational complexity can be reduced compared to the computational complexity of existing variations of GWO. DGWO was evaluated and compared to well-known swarm-based optimization algorithms using 30 CEC 2014 functions. In addition, the sensitivity of DGWO to its parameters was evaluated using 15 standard test functions. The comparative study and the sensitivity analysis for DGWO indicate that it provides competitive performance compared to the other tested algorithms. The source code of DGWO is available at: https://www.dropbox.com/s/2d16t46598u03y0/DistributedGreyWolfOptimizer.zip?dl=0.

KEYWORDS

Grey wolf optimizer, Island-based model, Distributed optimization algorithm, Optimization, Swarm-based optimization.

1. INTRODUCTION

Swarm-based optimization algorithms, such as the Bat algorithm [1]-[2], Cuckoo search [3]-[6], Whale optimization [7]-[9], Butterfly optimization [10]-[14], Grasshopper optimization [15], flower pollination [16] and Particle swarm optimization [1], [17], have been successfully used to solve difficult optimization problems in various fields (e.g., image processing [18]-[19], fuzzy logic [20]-[21], control engineering [22]-[25], scheduling problems [26]-[27]). The Grey Wolf Optimizer (GWO) is an interesting swarm-based optimization algorithm that was recently proposed to solve difficult optimization problems based on the hunting strategy and leadership hierarchy of grey wolves [28].

The GWO simulates the leadership hierarchy of grey wolves based on four hierarchical leaderships: alpha wolf, beta wolf, delta wolf and omega wolf. In addition, GWO simulates the hunting strategy of grey wolves based on three sequential steps: searching for prey, encircling prey and attacking prey. GWO has lately attracted much attention from the optimization community due to its attractive advantages. First, GWO is an efficient optimization algorithm that has a simple structure (Figure 1). Second, the simulation process of GWO is controlled by one key parameter (Section 2.1). Finally, GWO has been successfully tailored to solve various continuous optimization problems as well as discrete optimization problems (Section 2.3).

Like most of the swarm-based optimization algorithms, GWO may converge faster than expected to sub-optimal solutions [29]-[30]. This is because the evolutionary operators of GWO may not adequately preserve the diversity of the population over the course of the simulation process of GWO. Swarm-based optimization algorithms can be in general parallelized to run in different machines.

B. Abed-alguni and M. Barhoush are both with the Computer Sciences Department, Yarmouk University, Irbid, Jordan.
Email:Bilal.h@yu.edu.jo

Interestingly, the parallel swarm-based approach offers a possible solution to the problem of premature convergence of most of the swarm-based optimization algorithms [31]-[32]. This might be because the parallel approach allows the population of candidate solutions of a given optimization problem to be divided into several groups, which provides a better environment for unfit candidate solutions to evolve in each group.

The island model, which is a structured population model, can be integrated with the framework of a swarm-based optimization algorithm to facilitate its parallelization [21], [33]. Using the island model, the population of a parallel swarm-based optimization algorithm can be divided into *n* groups (islands), where a swarm-based optimization algorithm is applied independently to the population of each island. These islands periodically exchange selected candidate solutions among each other (i.e., migration process) in an attempt to maintain the diversity of population on each island.

The current paper introduces a Distributed Grey Wolf Optimizer (DGWO) that attempts to enhance the diversity of GWO by organizing its population into small islands based on the island model. DGWO applies the original GWO to the population of each island and then allows selected solutions to be exchanged among the islands based on the random ring topology and the best-worst migration policy.

The rest of the paper is organized as follows: Section 2 provides background information about the Grey Wolf Optimizer algorithm, the distributed island model and related work to the Grey Wolf Optimizer. Section 3 presents and discusses the Distributed Grey Wolf Optimizer algorithm. Section 4 presents the simulation results of the proposed algorithm and finally Section 5 presents the conclusions and future work.

2. PRELIMINARIES

This section briefly summarizes some of the underlying concepts of the grey wolf optimizer (Section 2.1) algorithm and the island model (Section 2.2). This section also provides an overview of recently proposed variations of grey wolf optimizer (Section 2.3).

2.1 Grey Wolf Optimizer

The Grey Wolf Optimizer (GWO) algorithm, which was developed by Mirjalili et al. [28], is an interesting nature-inspired optimization algorithm. GWO uses a simulation model based on the hierarchical leadership and hunting strategy of grey wolves. In the wild, grey wolves are top predators, which means that they are at the top of their food chain, with no natural predators. A pack of wolves is normally composed of 6 to 7 wolves, but it might have up to 15 wolves. In a wolf pack, normally an alpha (α) male and an alpha female wolves control the pack. The direct followers to the wolves are the beta (β) and delta (δ) wolves. The β and δ wolves help the α wolves to control and dominate the other wolves in the pack (omega (ω) wolves). The hunting strategy of grey wolves is a three-stage cooperative strategy (tracking and chasing prey, pursuing and encircling prey and attacking prey) [34].

Figure 1 shows the flow of the GWO algorithm. The first step of GWO is to generate a population of wolves (candidate solutions) $\vec{X}_i (i = 1,2, \ldots, N)$ for a given optimization problem. Each candidate solution is composed of *M* decision variables $\vec{X}_i = \{x_1, x_2, \ldots, x_M\}$. The fitness value of each candidate solution is calculated using the fitness function of the optimization problem to determine the hierarchical structure of the population. The solution with the best calculated fitness value is called the α solution ($\vec{X}_\alpha$), while the solutions with the second and third best fitness values are respectively called the β solution ($\vec{X}_\beta$) and δ solution ($\vec{X}_\delta$). The rest of the solutions in the population are called the ω solutions ($\vec{X}_\omega$) [20].

The improvement loop of GWO is composed of three stages: tracking and chasing prey, pursuing and encircling prey and attacking prey. Encircling prey is mathematically modelled in GWO as follows:

$$\vec{D} = \left|\vec{C}.\vec{X}_p\,(t) - \vec{X}\,(t)\right| \tag{1}$$

$$\vec{X}(t+1) = \vec{X}_p\,(t) - \vec{A}.\vec{D} \tag{2}$$

where *t* is the current iteration number, $\vec{A}$ and $\vec{C}$ are two coefficient vectors, $\vec{X}_p$ *is* the candidate solution that represents the prey and $\vec{X}\ (t)$ is the candidate solution that represents the grey wolf.

The two coefficient vectors $\vec{A}$ and $\vec{C}$ can be updated as follows:

$$\vec{A} = 2\vec{a}.\vec{r_1} - \vec{a} \tag{3}$$

$$\vec{C} = 2.\vec{r_2} \tag{4}$$

where $\vec{a}$ is a vector with values that linearly decrease from 2 to 0 over the course of the simulation of GWO and $\vec{r_1}$ and $\vec{r_2}$ are two random vectors between 0 to 1.

In GWO, the *ω* solutions are updated based on the *α, B* and *δ* solutions. Equations 5 to 11 represent the update process of the solutions:

$$\overrightarrow{D_\alpha} = |\vec{C}_1.\vec{X}_\alpha - \vec{X}| \tag{5}$$

$$\overrightarrow{D_\beta} = |\vec{C}_2.\vec{X}_\beta - \vec{X}| \tag{6}$$

$$\overrightarrow{D_\delta} = |\vec{C}_3.\vec{X}_\delta - \vec{X}| \tag{7}$$

$$\overrightarrow{X_1} = \vec{X}_\alpha - \overrightarrow{A_1}.(\overrightarrow{D_\alpha}) \tag{8}$$

$$\overrightarrow{X_2} = \vec{X}_\beta - \overrightarrow{A_2}.(\overrightarrow{D_\beta}) \tag{9}$$

$$\overrightarrow{X_3} = \vec{X}_\delta - \overrightarrow{A_3}.(\overrightarrow{D_\delta}) \tag{10}$$

$$\vec{X}(t+1) = \frac{\overrightarrow{X_1}+\overrightarrow{X_2}+\overrightarrow{X_3}}{3} \tag{11}$$

Approaching and attacking prey (exploitation stage) is simulated in GWO by decreasing the components of $\vec{a}$ from 2 to 0 over the course of simulation of GWO. According to Equation 3, the vector $\vec{a}$ controls the range of values of $\vec{A}$ which are in the range $[-2\vec{a},\ 2\vec{a}]$. It is worth noting that a candidate solution is updated in the direction of the best solutions (α, B and δ solutions) when $|\vec{A} < 1|$. The exploration of the search space is triggered in GWO using two settings. First, the candidate solutions diverge from the best solutions when $|\vec{A} > 1|$. Second, the components of $\vec{C}$, which are in the range [0, 2], provide weights for the best solutions in order to increase the influence of the best solutions $|\vec{C} > 1|$ or decrease their influence $|\vec{C} < 1|$ in Equation 1.

```
Initialize the population of n candidate solutions $\vec{X}_i (i = 1,2,\ldots,n)$
Initialize $\vec{A}$, $\vec{a}$ and $\vec{C}$
Calculate the fitness value of each candidate solution
$\vec{X}_\alpha$= the best candidate solution
$\vec{X}_\beta$= the second best candidate solution
$\vec{X}_\delta$= = the third best candidate solution
While (t < Max number of iterations)
for each candidate solution
        Update the values of the current candidate solution using Equation 11
end for
Update $\vec{A}$, $\vec{a}$ and $\vec{C}$
Calculate the fitness value of each candidate solution
Update $\vec{X}_\alpha$, $\vec{X}_\beta$ and $\vec{X}_\delta$
t=t+1
end while
return $\vec{X}_\alpha$
```

Figure 1. The Grey Wolf Optimizer (GWO) Algorithm.

It is important to note that GWO in its current form can be only applied directly to continuous optimization problems. GWO has been tailored for many real-world discrete optimization problems, such as feature selection [35], economic load dispatch problems [36]-[37] and scheduling problems [38]

-[39].

The computational complexity of GWO (Figure 1) can be calculated as follows:

1. Initializing the population of *n* candidate solutions requires *n* operations.
2. Initializing the parameters of GWO ($\vec{A}$, $\vec{a}$ and $\vec{C}$) requires 3 operations.
3. Calculating the first three best solutions ($\vec{X}_\alpha, \vec{X}_\beta$, and $\overrightarrow{X_\delta}$) requires *n* operations.
4. The following operations are conducted inside the while loop:
    a. The for loop that is used to update the population requires *n* operations.
    b. Calculating the fitness of each candidate solution in an island requires *n* operations.
    c. Updating $\vec{X}_\alpha, \vec{X}_\beta$ and $\vec{X}_\delta$ requires *n* operations.
    d. Updating the parameters of GWO ($\vec{A}$, $\vec{a}$ and $\vec{C}$) requires 3 operations.

    Overall, the while loop requires *m*.(*n* + *n*+ *3* +*1*) operations, where *m* is the maximum number of iterations. The number of operations can be further simplified to *m.n*
5. All the operations of the algorithm can be calculated as *n* + 3+ *n* + *m.n*, which can be simplified to *m.n,* because 2*n* is greater than 3 and *m.n* is greater than 2*n*.

In summary, the computational complexity of GWO is O(*m.n*). Note that any basic vector operation has been assumed to cost O(1) in the above analysis.

2.2 Island Model

The island model, which is a distributed population model, can be integrated with the framework of a swarm-based optimization algorithm to facilitate its parallelization [33]. Using the island model, the population of a swarm-based optimization algorithm can be divided into *n* groups (islands). Each island is assigned to a computation device, where a swarm-based optimization algorithm is applied to its population. An interaction process, called migration, is periodically triggered among the islands in the island model. In the migration process, selected candidate solutions are exchanged among islands in an attempt to maintain and amend the diversity of population on each island. The best-worst and random-random policies are the most used migration policies in the literature [31]-[33]. In the best-worst policy, the most fitted solutions in one island (say *m* solutions) are exchanged with the *m* worst fitted solutions in a neighbouring island. In the random- random policy, *m* random solutions in one island are swapped with *m* random solutions in a neighbouring island.

The islands in the island model are normally organized and arranged based on a given migration topology [40]. Star, random-star, mesh, random-mesh, ring and random-ring topologies are some popular migration topologies used with swarm-based algorithms [41]. The prefix "random" in the name of the migration topology, which indicates that the order of the islands in the topology changes each time the migration process is triggered.

Two parameters control the migration process among islands in the island model [42]. First, the migration frequency (M_f), which determines the number of iterations between two consecutive migration waves. Second, the migration rate (M_r), which is a parameter that determines the percentage of solutions to be exchanged between an island and a neighbouring island.

2.3 Variations of Grey Wolf Optimizer

Several hybrid GWO algorithms have been recently proposed in an attempt to control and amend premature convergence of GWO. Jayabarathi et al. [36] proposed a hybrid GWO algorithm that integrates the genetic operators (crossover and mutation) of the genetic algorithm (GA) into GWO to improve its exploration ability. The experimental results in [36] suggested that the hybrid GWO and GA algorithm provides good performance in solving several instances of the economic dispatch problem. However, the proposed hybrid algorithm requires heavy computations compared to the basic GWO. This is expected, because applying the genetic operators at each iteration of GWO for each candidate solution is a time-consuming process. Another disadvantage of the hybrid GWO algorithm is that it has a complex structure compared to the GWO algorithm. Tawhid and Ali [29] integrated the whole GA algorithm into GWO to solve the minimization problem of the energy function of the molecule. Although the hybrid GA and GWO algorithm performs much better than the basic GWO, it requires more computational time than GWO. Jitkongchuen [43] proposed a hybrid swarm-based algorithm, between the differential evolution (DE) algorithm and GWO, for solving numerical optimization functions. The simulation results in [43] suggested that the hybrid GWO is more reliable and more accurate in solving difficult numerical optimization problems than DE, self-adaptive DE and particle swarm optimization (PSO). Unfortunately, the hybrid GWO and DE algorithm is complex and requires heavy computations compared to GWO. Zawbaa et al. [44] combined Antlion optimization (ALO) and GWO in one algorithm (ALO-GWO). ALO-GWO was particularly designed to solve the feature selection problem in large datasets. The simulation results in [44] indicated that ALO-GWO provides good performance in solving feature extraction problems in large datasets compared to PSO and GA. However, ALO-GWO may require a long processing time as most of the hybrid optimization algorithms.

Various intelligent techniques have been successfully used with GWO to enhance its convergence behaviour. Saremi et al. [45] suggested the EPD-GWO algorithm that uses the evolutionary population dynamics (EPD) technique to improve the diversity of population in GWO. In other words, EPD is used in EPD-GWO to improve the exploration of candidate solutions. EDP repositions the worst candidate solutions in the population into the neighbourhoods of the alpha, beta, delta and omega wolves. The simulation results in [45] indicated that EPD-GWO provides better results than GWO. However, the computational complexity of EPD-GWO is higher than the computational complexity of GWO because EPD has to be repeated at each iteration of EPD-GWO. Rodríguez et al. [46] proposed a dynamic variation of GWO that dynamically tunes the parameters of GWO during the simulation process of GWO in order to obtain the best possible performance out of GWO. However, manipulating the parameters of GWO as in [46] does not provide significant enhancement in the performance than the original GWO algorithm. The enhanced GWO (EGWO) algorithm, that was proposed by Joshi and Arora [47], is an another adaptive variation of GWO. EGWO dynamically changes the values of the key parameter of GWO ($\vec{a}$) over the course of its simulation. Moreover, EGWO enhances the exploitation mechanism of GWO by making the best use of the best solution (alpha solution). The experimental results of EGWO compared to standard optimization algorithms (PSO, Firefly Algorithm (FA) and Flower Pollination Algorithm (FPA)) proves that EGWO is a competitive algorithm for solving constrained optimization problems. Malik et al. [48] introduced the wdGWO algorithm which combines the weighted distance (wd) technique with GWO. In wdGWO, the update strategy of the candidate solutions in GWO is modified and the average function of best solutions is replaced with the weighted sum of best solutions. According to the experimental results in [48], the wdGWO showed superior performance compared to basic optimization algorithms (FA, Artificial bee colony, Cuckoo search and PSO).

Moreover, Emary et al. [49] introduced the experienced Grey Wolf Optimizer (EGWO) algorithm that incorporates reinforcement learning (RL) and artificial neural networks (ANNs) into GWO to improve its performance. EGWO uses RL to update the parameters of each candidate solution in the population of GWO. EGWO uses ANNs to estimate the expertness of each candidate solution. The expertness estimation of a solution is used to control its exploration rate. EGWO was evaluated and compared to three optimization algorithms (GWO, PSO, GA). Joshi and Arora [50] proposed an enhanced GWO (E-GWO) algorithm that uses an improved hunting mechanism to balance between the exploration and exploitation of candidate solutions in GWO. Kohli and Arora [51] proposed the chaotic GWO (CGWO) algorithm that incorporates the chaos theory into GWO in an attempt to improve the convergence behaviour of GWO for constrained optimization problems. In CGWO, several types of chaotic maps are employed to adjust the main parameter of GWO ($\vec{a}$). The simulation results of CGWO compared to well-known algorithms (PSO, Firefly Algorithm (FA) and Flower Pollination Algorithm (FPA)) suggest that CGWO provides good results compared to the other algorithms. Heidari and Pahlavani [52] introduced a modified GWO that uses the greedy selection method and the Lévy flight operator with GWO in an attempt to improve the exploration mechanism of GWO. The simulation results in [52] indicated that the improved GWO is more reliable and more effective in solving both discrete and continuous optimization problems than popular state-of-the-art swarm-based optimization algorithms. However, the greedy selection method is not the best selection mechanism according to Abed-alguni and Alkhateeb [4]. Moreover, using the greedy selection method in an optimization algorithm may cause premature convergence [4], [53]. Gupta and Deep [54]-[55] presented an improved GWO algorithm called RW-GWO that uses random walk to enhance the search ability of grey wolves. According to the simulation results in [54]-[55], RW-GWO shows high efficiency in solving both continuous and discrete optimization algorithms.

In summary, the computational complexity of most of the hybrid GWO algorithms (e.g. hybrid GA and GWO [29], [36], DE-GWO [43], EGWO [49], ALO-GWO [44]) is much higher than the computational complexity of the original GWO. This is because the integrated search method in GWO is normally repeated at each iteration of GWO. Moreover, hybrid GWO algorithms have complex structures compared to the basic GWO algorithm. Other enhanced versions of the GWO algorithm (e.g. EPD-GWO [45], GWO and Lévy flight operator [52], Dynamic GWO [46], wdGWO [48], EGWO [47]) provide insignificant enhancement in the performance compared to GWO or can be applied only to specific applications. In the next Section, a distributed variation of GWO (DGWO) is introduced in an attempt to enhance the diversity of GWO by organizing its population into small islands based on the island model. An interesting feature about DGWO is that it can run in parallel devices, which means that its computational complexity can be reduced compared to the computational complexity of hybrid GWO algorithms. However, DGWO can be directly applied to continuous optimization problems. DGWO should be tailored to be applicable to real-world discrete optimization problems.

3. Distributed Grey Wolf Optimizer

The Grey Wolf Optimizer (GWO) is a nature-inspired optimization algorithm that mimics the hunting strategy and leadership hierarchy of grey wolves [28]. GWO has been applied successfully to various continuous optimization problems [45], [48], [52] and discrete optimization problems [29], [36], [48], [52]. A problem with GWO is that its improvement loop may not maintain the diversity of its population due the imperfection of its evolutionary operators. Such a problem is a common problem with all optimization algorithms.

The island model enhances the performance and run-time of optimization algorithms. It also provides better chances for unfit solutions in each island to evolve and improve. The distributed genetic algorithm [56], distributed differential evolution [57]-[58], distributed particle swarm optimization [59] and distributed ant colony [60] algorithms are but few examples of successful island-based optimization algorithms.

The current section introduces a Distributed Grey Wolf Optimizer (DGWO) algorithm in an attempt to improve the diversity of GWO by organizing its population into small islands based on the island model. DGWO applies the original GWO to the population of each island and then allows selected solutions to be exchanged among the islands based on the random-ring topology and the best-worst migration policy.

Figure 2 shows the pseudo code of the DGWO algorithm. The first step of DGWO is to determine the total number of candidate solutions (n) for a given number of islands (s) and the maximum number of iterations (*MaxItr)* of DGWO. The second step is to initialize the parameters of the island model. The next step is to generate k candidate solutions for each island and then to calculate the number of migration waves (M_w) and number of migrant solutions (n_r).

DGWO generates k candidate solutions for each island before the beginning of the evolution process of DGWO. In DGWO, the evolution process of the basic GWO algorithm is synchronously applied to each individual island. After each M_f iterations (migration frequency), a number of candidate solutions are swapped between each two neighbouring islands based on the random-ring topology and the best-worst migration policy. In the random-ring topology, the neighbouring relationships are unidirectional relationships (Figure 3). However, the neighbouring relationships in the random-ring topology change after each migration wave among the islands. The number of candidate solutions to be exchanged among the islands is specified by the migration rate (M_r).

The migration process among islands takes place each time the maximum number of iterations that is specified by M_f is reached. Before the beginning of the migration process, the islands in DGWO are organized to form a unidirectional ring based on the principles of the random ring topology. The most fitted solutions in one island (say m solutions) are exchanged with the m worst fitted solutions in a neighbouring island based on the best-worst migration policy. Let $ppo_i = \{x_1^i, x_2^i, \dots, x_s^i\}$ and $ppo_j = \{x_1^j, x_2^j, \dots, x_s^j\}$ be the neighboring islands I_i and I_j, respectively. If we assume that R_m = 20%, n=150 and s= 5, the number of migrant solutions is R_m × (n/s) = 20%×(150/5) = 6. Let ppo_i and ppo_j be two lists ordered in ascending order based on their objective values, where $f(x_1^i) \leq f(x_2^i) \leq \cdots \leq f(x_s^i)$ and $f(x_1^j) \leq f(x_2^j) \leq \cdots \leq f(x_s^j)$. If we assume that there is a unidirectional edge from I_i to I_j, the first six candidate solutions in I_i $(x_1^i, x_2^i, x_3^i, x_4^i, x_5^i, x_6^i)$ will be exchanged with the last six solutions in I_j $\{x_{s-5}^j, x_{s-4}^j, x_{s-3}^j, x_{s-2}^j, x_{s-1}^j, x_s^j\}$.

```
Determine the total number of candidate solutions of all islands (n) and the maximum number of iterations (MaxItr).
Initialize the parameters of the island model (number of islands (s), migration frequency ($M_f$), migration rate ($M_r$))
Calculate the population size for each island (k=n/s)
Calculate the number of migration waves ($M_w$= MaxItr/ $M_f$)
Calculate the number of migrant solutions ($n_r$=$n/s \times M_r$)
Initialize the population of k candidate solutions for each island $\vec{X}_i^j$ $(i = 1,2,\dots,k)$ and $(j = 1,2,\dots,s)$
for $i = 1$ to $M_w$ .................................(1)
    for $j = 1$ to $s$...........................................(2)
        Initialize $\overrightarrow{A}^j$, $\vec{a}^j$ and $\vec{C}^j$
        t=0
        Calculate the fitness value of each candidate solution
        $\vec{X}_\alpha^j$= the best candidate solution in island j
        $\vec{X}_\beta^j$= the second best candidate solution in island j
        $\vec{X}_\delta^j$ = the third best candidate solution in island j
        While (t< $M_f$).............................................................(3)
            for each candidate solution in island j .................................(4)
                Update the values of the current candidate solution using Equation 11
            end for
            Update $\vec{A}^j$, $\vec{a}^j$ and $\vec{C}^j$
            Calculate the fitness value of each candidate solution in island j
            Update$\vec{X}_\alpha^j$ ,$\vec{X}_\beta^j$ and $\vec{X}_\delta^j$
            Replace the nr best candidate solutions in island j with the $n_r$ worst candidate solutions in island ((j+1) mod s)
            t=t+1
        end while
    end for
end for
Calculate $\vec{X}_\alpha$($\vec{X}_\alpha^j$ with the best fitness)
return the $\vec{X}_\alpha$
```

Figure 2. The Distributed Grey Wolf Optimizer (DGWO) Algorithm.

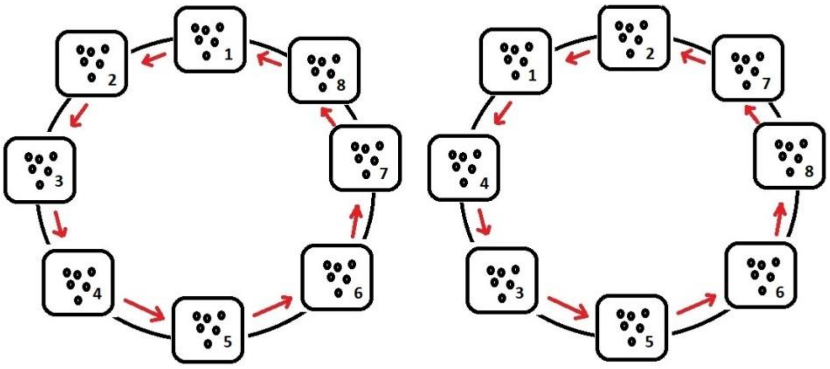


Figure 3. Random-ring Migration Topology.

The computational complexity of DGWO (Figure 2) can be calculated based on its main steps as follows:

1. The first five steps require 5 operations.
2. Initializing the population of n candidate solutions for all islands requires n operations.
3. Calculating the fitness of all candidate solutions requires n operations.
4. Calculating the first three best solutions in an island requires k operations.
5. The most inner for loop (number 4) that is used to update the population of an island requires k operations.
6. Calculating the fitness of each candidate solution in an island requires k operations.
7. Updating the first three best solutions in an island requires k operations.
8. Replacing the n_r best candidate solutions in an island with the n_r worst candidate solutions in another island requires $k+ k+ n_r$ operations, which can be simplified to k because $k > n_r$.
9. Overall, the while loop (number 3) requires $M_f.(k + k + k + n_r)$ operations. This step can be simplified to $M_f.\ k$, because $k>n_r$.
10. For loop number 2 requires $s.(k + k + M_f.\ k)$ operations, which can be simplified to $s.M_f.k$, because $s.M_f.k$ is larger than $s.k$.
11. For loop number 1 requires $M_w.\ s.\ M_f.\ k$ operations.
12. The last step that calculates the best candidate solution in all islands requires s operations.
13. All the operations of the algorithm can be calculated as $5+2n+k+M_w.s.M_f.k+s$, which can be simplified to $M_w.s.M_f.k$ operations.

In summary, the computational complexity of DGWO is $O(M_w.s.M_f.k)$. On the other hand, the computational complexity of the basic GWO is $O(m.n)$, where m is the maximum number of iterations and n is the number of candidate solutions. Thus, it is better to run DGWO in parallel devices which will reduce the computational complexity of DGWO to $O((M_w.s.M_f.k)/s)$, where s is the number of parallel devices (number of islands). In this case, the complexity of DGWO can be simplified to $O(M_w\ .M_f\ .k)$. Note that any basic vector operation has been assumed to cost O(1) in the above analysis.

4. Experiments

In this section, the DGWO is benchmarked on 15 test functions (Table 1 and Table 2). These functions are standardtest functions used widely by the researchers to evaluate and compare the performance of swarm-bsed evolutionary algorithms [3]-[4], [28], [43], [48], [61]. Section 4.1 shows the experimental setup. Section 4.2 provides an analysis of the performance of DGWO based on different experimental scenarios. Section 4.3 provides analysis about the performance of DGWO compared to recently proposed optimization algorithms (Grey Wolf Optimizer (GWO) [28], Cuckoo search (CS) [3], adaptive differential evolution with linear population size reduction evolution (L-SHADE) [62], memory-based hybrid Dragonfly (MHDA) [63] and Fireworks algorithm with differential mutation (FWA-DM) [64].

Table 1. Test functions.

Abbreviation	Function name	Range	D	$f(\vec{X}^*)$
f_1	Generalized Schwefel's Problem 2.26	[-500,500]	30	−418.983 × D
f_2	Griewank's Function	[-10,10]	30	0
f_3	Whitley's Function	[-10,10]	30	0
f_4	Ackley's Function 2.9	[32.768, 32.768]	30	0
f_5	Alpine's Function	[-10,10]	30	0
f_6	Schaffer's Function	[-100,100]	2	0
f_7	Rastrigin's Function 2.5	[-5.12, 5.12]	30	0
f_8	Inverted Cosine Wave Function	[-5, 5]	30	-D+1
f_9	Levy Function	[-10, 10]	30	0
f_{10}	Schwefel's 2.22 Function	[-100,100]	30	0
f_{11}	Rotated Hyper-ellipsoid Function	[-65.536, 65.536]	30	0
f_{12}	Shifted Sphere Function	[-100,100]	30	0
f_{13}	Shifted Schwefel Function	[-100,100]	30	0
f_{14}	Shifted Rastrigin’s Function	[-5, 5]	30	-330
f_{15}	Shifted Expanded Griewank’s Plus	[-5, 5]	30	-130

Table 2. Mathematical formulae of test functions.

Formula
$f_1(X) = -\sum_{i=1}^{n}[x_i \sin(\sqrt{\lvert x_i \rvert})]$
$f_2(X) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos(\frac{x_i}{\sqrt{i}}) + 1$
$f_3(x_1 \cdots x_n) = \sum_{i=1}^{n} \sum_{j=1}^{n} (\frac{(100(x_i^2 - x_j)^2 + (1 - x_j)^2)^2}{4000} - cos(100(x_i^2 - x_j)^2 + (1 - x_j)^2) + 1)$
$f_4(X) = -20\, e^{(-0.2\times\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2})} - e^{[\frac{1}{n}\sum_{i=1}^{n} \cos(2\pi x_i)]} + 20 + e^{(1)}$
$f_5(X) = \sum_{i=1}^{n} \lvert x_i \sin(x_i) + 0.1 x_i \rvert$
$f_6(x, y) = 0.5 + \frac{sin^2(x^2 + y^2)^2 - 0.5}{(1 + 0.001(x^2 + y^2))^2}$
$f_7(X) = \sum_{i=1}^{n}[x_i^2 - 10\cos(2\pi x_i) + 10]$
$f_8(X) = -\sum_{i=1}^{n-1}\{ e^{\left[\frac{-(x_i^2 + x_{i+1}^2 + 0.5 x_i x_{i+1})}{8}\right]} \cos(4 \times \sqrt{x_i^2 + x_{i+1}^2 + 0.5 x_i x_{i+1}})\}$
$f_9(x) = sin^2(\pi w_1) + \sum_{i=1}^{d-1}(w_i - 1)^2[1 + 10 sin^2(\pi w_i + 1)] + (w_d - 1)^2[1 + sin^2(2\pi w_d)],$ where $w_i = 1 + \frac{x_i - 1}{4}$, for all *i* = 1, 2, ..., *d*
$f_{10}(X) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$
$f_{11}(X) = \sum_{i=1}^{n} (\sum_{j=1}^{i} x_j)^2$
$f_{12}\,(X) == \sum_{i=1}^{D} z_i^2 + f_{bias} = \pi,$ where ***z*=*X*-*o*** and *f_{bias}*=-450
$f_{13}(x) = \sum_{i=1}^{D} (\sum_{j=1}^{i} z_j)^2 + f\_bias,$ where ***z*=*X*-*o*** and f_{bias}=-450
$f_{14}(x) = \sum_{i=1}^{D}(z_i^2 - 10\cos(2\pi z_i) + 10) + f\_bias,$ where ***z*=*X*-*o*** and f_{bias}=-330
The definition of f_{15} is given in [66], where ***z*=*X*-*o*** f_{bias}=-130

4.1 Experimental Setup

In each algorithm, the maximum number of iterations was 10,000 and the size of population was 30. For the GWO and DGWO algorithms, the values of vector $\vec{a}$ were linearly decreased form 2 to 0 over the course of their simulation process. The parameters of the island model in DGWO were tested for different values as shown in Table 3. The parameters of each test function are shown in Tables 1 and 2. The percentage of abandonment in CSwas 25% as suggested in [4], [66]. The parameters of L-SHADE (external archive size, historical memory size, control parameter) were dynamically tuned as in [62]. The parameter setting of MHDA was taken from refernce [63]. The parameters of FWA-DM were set as follows: A_{init} (initial amplitude)= 0.2, A_{final} (final amplitude)= 0.001, *F* (scaling factor)= 0.5 and *CR* (Crossover Operator)= 0.9 as in [64].

4.2 Analysis of the Parameters of DGWO

Table 3 shows nine experimental scenarios used to investigate the relationships between the performance of DGWO and the parameters of the island model (*s*, M_f, M_r). The purpose of the first three scenarios is to investigate the relationship between the island number *s* and the performance of DGWO. Scenarios 3-6 aim to investigate the influence of the migration frequency M_f on the performance of DGWO. The purpose of the last three scenarios is to study the influence of the migration rate M_r on the performance of DGWO.

Table 3. Scenarios for analyzing the parameters of DGWO.

Scenario	***s***	**M_f**	**M_r%**
Scenario 1	2	100	20
Scenario 2	5	100	20
Scenario 3	10	100	20
Scenario 4	10	50	20
Scenario 5	10	100	20
Scenario 6	10	500	20
Scenario 7	10	50	10
Scenario 8	10	50	20
Scenario 9	10	50	30

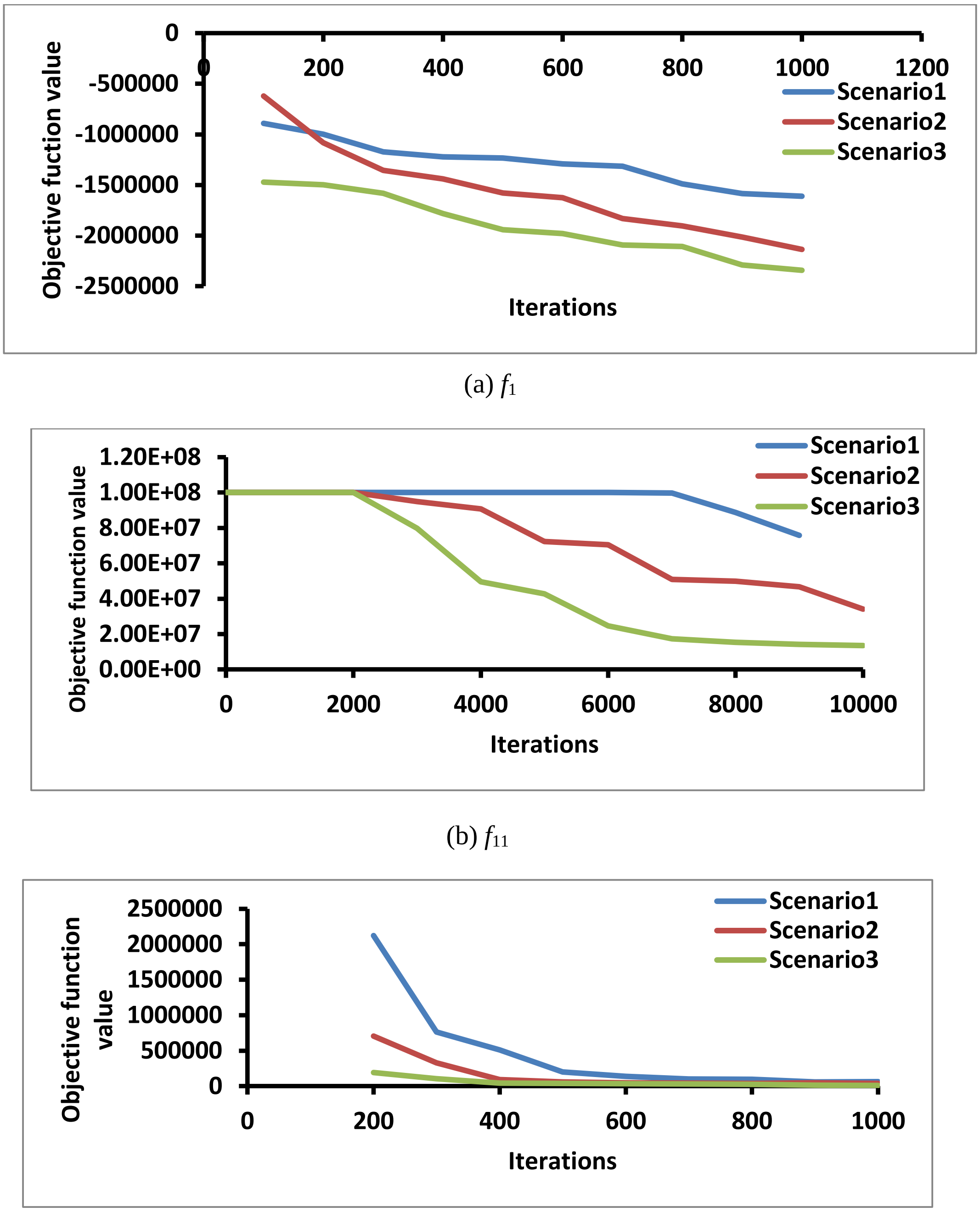
Tables 4-6 show the simulation results of the nine experimental scenarios ilustrated in Table 3. The best results in the tables (lowest objective values) are highlighted with **bold**. The results were averged over 50 independent runs.

Table 4. Simulation results of the algorithms for 15 test functions, D=30, runs=50, iterations=10,000 (Scenario 1, Scenario 2, Scenario 3).

Function	**Scenario 1** ***s* = 2**	**Scenario 2** ***s* = 5**	**Scenario 3** ***s* = 10**
f_1	-1.56E+06	-2.08E+06	**-2.10E+06**
f_2	0.00E+00	0.00E+00	0.00E+00
f_3	4.57E+01	4.52E+01	**4.49E+01**
f_4	1.98E+01	**4.44E-16**	**4.44E-16**
f_5	6.44E-220	**1.97E-222**	1.64E-210
f_6	5.00E-01	5.00E-01	5.00E-01
f_7	0.00E+00	0.00E+00	0.00E+00
f_8	-9.00E+00	-9.00E+00	-9.00E+00
f_9	1.32E+00	1.32E+00	**8.52E-01**
f_{10}	7.27E+03	1.15E+04	**4.21E+03**
f_{11}	8.17E+07	5.27E+07	**1.36E+07**
f_{12}	1.00E+08	1.00E+08	1.00E+08
f_{13}	1.00E+08	1.00E+08	1.00E+08
f_{14}	-2.33E+02	-2.38E+02	**-2.40E+02**
f_{15}	2.49E+04	3.52E+03	**3.15E+03**

Table 4 shows the simulation results of DGWO under different island sizes (*s*= 2, *s*=5, *s*=10). The results in the table show that DGWO in scenario 3 (*s*=10) performed better than in scenario 1 (*s*=2) and scenario 2 (*s*=5). The results clearly indicate that the performance of DGWO improves with an increase in the number of islands. This is expected, because the existence of many islands (large value of *s*) allows different search regions to be explored simultaneously, while the existence of few islands (low value of *s*) means that few number of search regions can be explored simultaneously [31]-[32]. In addition, the population in a small island would most likely converge to suboptimal solutions earlier than expected [42]. It is worth pointing out that choosing a large value of *s* increases the computational complexity of DGWO. For example, if the number of islands is 10, then GWO will be repeated 10 times at each iteration of DGWO. Thus, it is better to run DGWO in parallel devices to reduce its computational complexity. Based on the results in Table 4, *s*=10 was used in Tables 5 and 6.

The convergence curves of the first three scenarios of DGWO for three functions (f_1, f_{11}, f_{15}) are shown in Figure 4. It is obvious that convergence increases with the increase in iterations for all functions. Figure 4(a), Figure 4(b) and Figure 4(c) show that DGWO in scenario 3 is the fastest converging algorithm.



(a) f_1

(b) f_{11}

(c) f_{15}

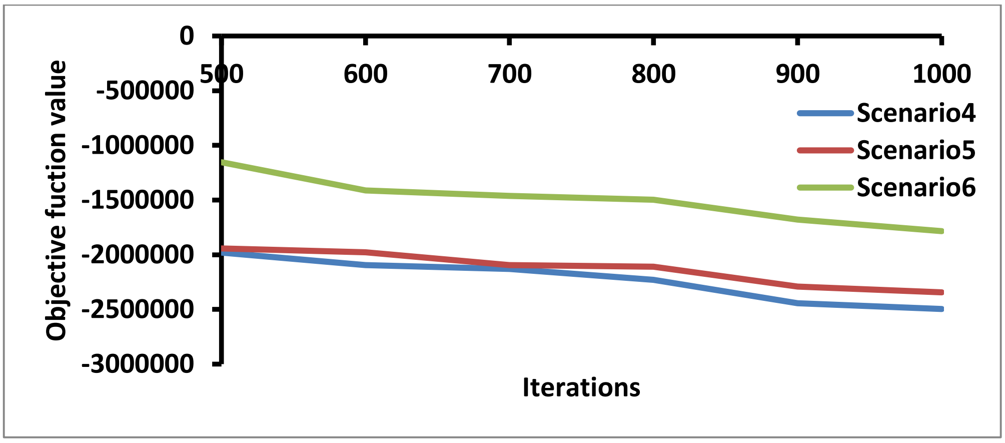
Figure 4. Convergence Plots of DGWO for Scenarios 1-3.

Table 5 shows the simulation results of DGWO under different migration frequencies ($M_f = 50$, $M_f = 100$, $M_f = 500$). It is obvious that DGWO in scenario 4 ($M_f = 50$) performed the best followed by DGWO in scenario 2 ($M_f = 100$) and then DGWO in scenario 3 ($M_f = 500$). These results suggest that low migration frequencies improve the performance of DGWO compared to high migration frequencies. Basically, low migration frequencies provide more chances for a reasonable number of candidate solutions from the source island to move to the destination island with limited effect on the candidate solutions in the destination island. Consequently, the likelihood that convergence will take longer time to occur increases [31]-[32]. Based on the illustrated results in Table 5, $M_f = 50$ was used in Table 6.

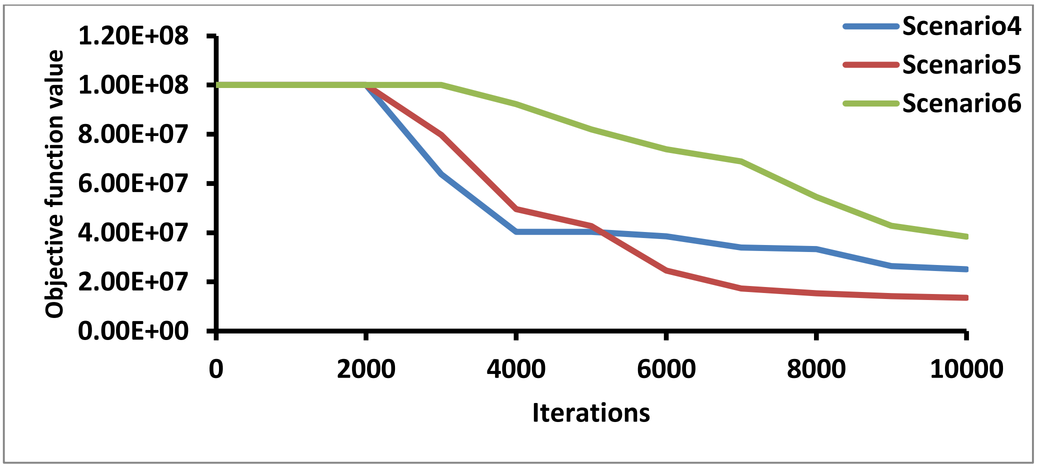
Table 5. Simulation results of the algorithms for 15 test functions, D=30, runs=50, iterations=10,000 (Scenario 4, Scenario 5, Scenario 6).

Function	**Scenario 4 (M_f = 50)**	**Scenario 5 (M_f = 100)**	**Scenario 6 (M_f = 50)**
f_1	**-2.27E+06**	-2.10E+06	-2.13E+06
f_2	0.00E+00	0.00E+00	0.00E+00
f_3	4.49E+01	4.49E+01	**4.47E+01**
f_4	4.44E-16	4.44E-16	4.44E-16
f_5	2.38E-224	**1.64E-221**	2.74E-218
f_6	5.00E-01	5.00E-01	5.00E-01
f_7	0.00E+00	0.00E+00	0.00E+00
f_8	-9.00E+00	-9.00E+00	-9.00E+00
f_9	**7.26E-01**	8.52E-01	1.57E+03
f_{10}	4.90E+03	**4.21E+03**	9.74E+03
f_{11}	1.40E+07	**1.36E+07**	7.76E+07
f_{12}	**9.49E+06**	1.00E+08	1.00E+08
f_{13}	1.00E+08	1.00E+08	1.00E+08
f_{14}	**-2.60E+02**	-2.40E+02	-2.44E+02
f_{15}	**1.10E+03**	3.15E+03	9.01E+03

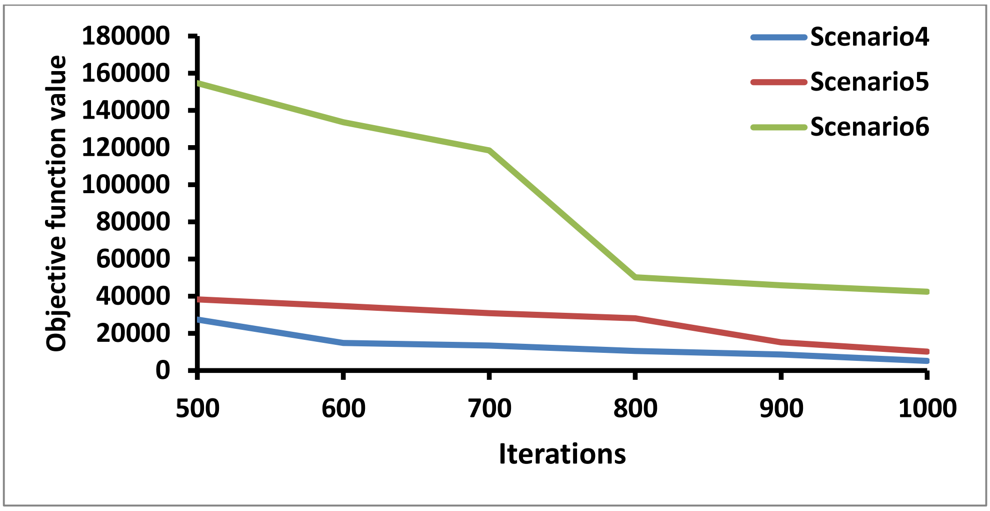
Figure 5 shows the convergence curves of the second three scenarios of DGWO for three functions (Figure 5(a) (f_1,) Figure 5(b) (f_{11}), Figure 5(c) (f_{15})). Figure 5(a) and Figure 5(c) show that DGWO in scenario 4 converges faster than the other algorithms, while Figure 5(b) shows that DGWO in scenario 5 converges faster to a solution compared to the other algorithms.



(a) f_1



(b) f_{11}



(c) f_{15}

Figure 5. Convergence Plots of DGWO for Scenarios 4-6.

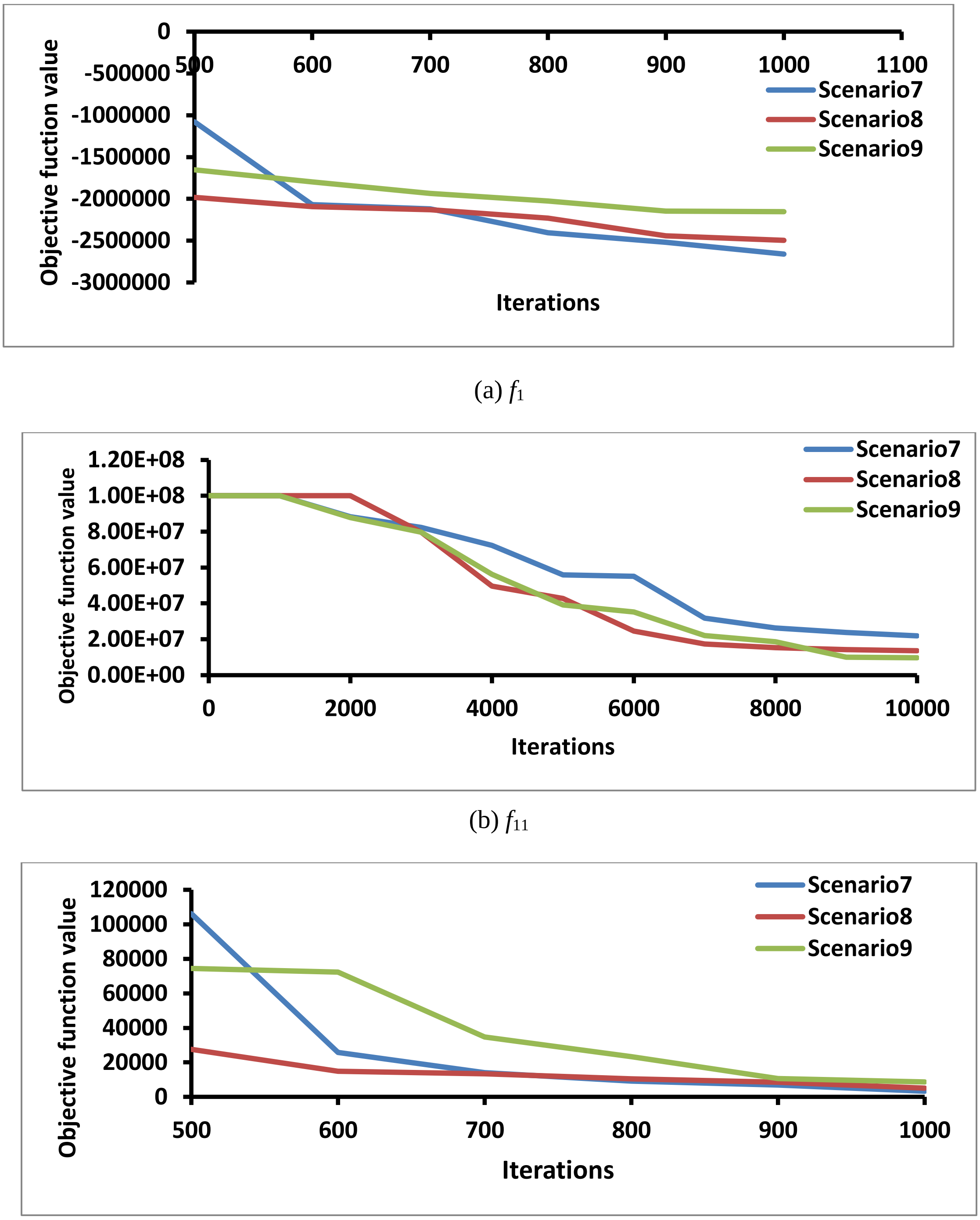
Table 6 shows the simulation results of DGWO under different migration rates (M_r = 10%, M_r = 20%, M_r= 30%). It can be clearly observed that DGWO in scenario 8 (M_r = 20%) performed better than DGWO in scenario 7 (M_r = 10%) and DGWO in scenario 9 (M_r = 30%). These results suggest that there is no clear indication whether high values or low values of M_r improve the performance of DGWO. A possible explanation for the results is that replacing a large number of candidate solutions in an island has an unclear effect on the diversity of its population, while replacing few candidate solutions in an island can act as a seed to enhance its diversity but with a limited effect [31]-[32].

Table 6. Simulation results of the algorithms for 15 test functions, D=30, runs=50, iterations=10,000 (Scenario 7, Scenario 8, Scenario 9).

Function	**Scenario 7** ***M***	**Scenario 8**	**Scenario 9**
	M_r = 10%	**M_r = 20%**	**M_r = 30%**
f_1	**-2.75E+06**	-2.27E+06	-2.19E+06
f_2	0.00E+00	0.00E+00	0.00E+00
f_3	4.26E+01	4.49E+01	**4.12E+01**
f_4	4.44E-16	4.44E-16	4.44E-16
f_5	2.24E-219	**2.38E-224**	1.83E-223
f_6	5.00E-01	5.00E-01	5.00E-01
f_7	0.00E+0	0.00E+0	0.00E+0
f_8	-9.0E+00	-9.0E+00	-9.0E+0
f_9	8.52E-01	**7.26E-01**	1.32E+0
f_{10}	1.72E+04	**4.90E+03**	1.65E+04
f_{11}	2.91E+07	1.40E+07	**3.15E+06**
f_{12}	1.00E+08	**9.49E+06**	4.52E+07
f_{13}	1.00E+08	1.00E+08	1.00E+08
f_{14}	-2.54E+02	**-2.6E+02**	-2.46E+02
f_{15}	**1.00E+03**	1.10E+03	1.14E+03

Figure 6 shows the convergence curves of the last three scenarios of DGWO for three functions (Figure 6(a) (f_1,) Figure 6(b) (f_{11}), Figure 6(c) (f_{15})). Figure 6(a) and Figure 6(c) show that DGWO in scenario 7 converges faster than the other algorithms, while Figure 6(b) shows that DGWO in scenario 9 converges faster to a solution compared to the other algorithms. Note that DGWO in Scenario 8 achieved the second best convergence speed for the three functions.

In conclusion, the overall experimental results in this section indicate that high values of s and low values of M_f improve the performance of DGWO. However, there is no clear indication whether high values or low values of M_r improve the performance of DGWO.



(a) f_1

(b) f_{11}

(c) f_{15}

Figure 6. Convergence Plots of DGWO for Scenarios 7-9.

4.3 Comparison among DGWO and Other Algorithms

The single-objective real-parameter optimization-benchmark suit of CEC2014 is composed of 30 functions (Table 7). This suit represents an approximation of real-world optimization problems. The search range of each function in the suit is $[-100.100]^D$. More details about these functions are available in [67].

Using the single-objective real-parameter optimization-benchmark suit of CEC2014 with 30 dimensions (30 decision variables) [67], the performance of DGWO (Scenario 8) was compared with the performance of GWO and recently proposed optimization algorithms: Cuckoo search (CS), adaptive differential evolution with linear population size reduction evolution (L-SHADE), memory-based hybrid Dragonfly algorithm (MHDA) and Fireworks algorithm with differential mutation (FWA-DM).

Table 7. Single-objective real-parameter optimization-benchmark suit of CEC2014.

Function	**Function Type**
f_1- f_3	Unimodal functions
f_4- f_{16}	Multimodal functions
f_{17}- f_{22}	Hybrid functions
f_{23}- f_{30}	Composite functions

Table 8 shows the function error value (FEV) for the 30CEC2014 functions. The FEVis the distance between theaverage of best objective values found in all runs and the true optimal value. Note that the lowest FEV for each function (best result) is marked with **Bold**. The simulation results in Table 8show that L-SHADE ouperforms the other optimization algorithms by providing the lowest FEV for 11 functions of the 30 functions. This is expected, because L-SHADE is a dynamic differential evolution algorithm thatdynamically adjustsitsinternal parameters and population size over the course of its iterations. Interestingly, DGWO is the second best performing optimization algorithm by producing the lowest FEV for 10 functions of the 30 test functions. This is because DGWO synchronously applies GWO to multiple islands, which accelerates its convergence to a good solution.

Table 8. Simulation results of DGWO compared to five optimization algorithms. D= 30, runs=50, number of iterations is 10,000.

Function	**GWO**	**DGWO (Scenario 8)**	**CS**	**L-SHADE**	**MHDA**	**FWA-DM**
f_1	2.00E+03	4.36E+00	3.47E + 07	**9.00E-15**	3.59E+03	4.91E+05
f_2	2.12E+03	2.36E+00	2.50E + 07	8.50E-11	3.82E+03	**2.50E-16**
f_3	2.89E-01	2.54E-04	4.10E + 04	5.83E-10	5.80E-07	**1.88E-16**
f_4	8.75E-03	**1.63E-09**	4.22E+02	2.58E-09	1.42E-08	2.23E+01
f_5	3.96E+02	2.00E+02	5.00E+01	2.00E+01	**2.36E+00**	2.11E+01
f_6	5.19E+01	1.21E+00	3.63E+01	1.25E-06	**8.52E-14**	1.82E+01
f_7	2.89E-03	8.53E-10	1.86E+00	7.25E-09	**2.25E-11**	2.53E-03
f_8	1.33E+00	**1.51E-19**	3.89E+02	1.25E-09	2.20E-19	9.53E-15
f_9	1.82E+01	**1.03E+00**	3.00E+03	8.96E+00	5.30E+00	6.54E+01
f_{10}	9.79E+00	**3.20E-03**	4.37E+03	2.36E-02	1.22E+03	1.13E+01
f_{11}	1.99E+04	2.95E+03	4.00E+03	2.30E+03	**1.52E+02**	2.19E+03
f_{12}	8.50E+00	**6.30E-02**	4.78E-01	9.00E-01	1.42E-01	3.25E-01
f_{13}	2.19E+00	4.59E-01	4.81E-01	6.50E-01	4.78E-01	**3.11E-01**
f_{14}	2.35E-01	**1.99E-01**	4.28E-01	8.60E-01	5.43E-01	2.99E-01
f_{15}	1.01E+02	7.23E+01	9.94E+01	**1.60E+00**	3.25E+00	8.36E+00
f_{16}	1.90E+01	**9.53E+00**	1.53E+01	1.02E+01	1.06E+01	1.10E+01
f_{17}	1.66E+02	4.55E+03	3.47E+06	**2.20E+00**	4.53E+02	6.59E+03
f_{18}	8.77E+00	3.94E+01	3.90E+03	**1.90E+00**	3.69E+00	7.24E+01
f_{19}	4.96E+01	1.22E+02	6.14E+01	**5.30E+00**	3.78E+02	1.04E+01
f_{20}	6.20E+01	4.73E+02	3.97E+04	**4.30E+00**	7.09E+02	4.37E+01
f_{21}	1.04E+03	7.09E+02	3.57E+05	3.69E+02	**2.57E+02**	8.75E+02
f_{22}	2.42E+02	2.73E+02	9.47E+02	**1.32E+02**	2.73E+02	1.62E+02
f_{23}	3.65E+02	**3.69E+01**	3.78E+02	3.26E+02	3.10E+03	3.16E+02
f_{24}	2.24E+02	2.25E+02	2.89E+02	**1.93E+02**	2.26E+02	2.96E+02
f_{25}	2.45E+02	2.11E+02	3.26E+02	**2.00E+02**	2.11E+02	2.09E+02
f_{26}	3.29E+02	2.10E+02	2.22E+02	2.69E+02	1.00E+02	**9.93E+01**
f_{27}	2.95E+02	4.09E+02	5.22E+02	**1.26E+02**	4.05E+02	4.10E+02
f_{28}	5.36E+02	1.65E+03	3.86E+03	**3.62E+02**	1.54E+03	4.22E+02
f_{29}	2.39E+02	**2.29E+02**	2.59E+05	7.33E+02	7.86E+02	2.78E+02
f_{30}	3.32E+02	**2.83E+00**	2.39E+04	6.99E+02	2.63E+03	4.69E+02

This indicates that DGWO performs well compared to powerful optimization algorithms. Note that GWO and CS have the worst performance compared to the other algorithms. A possible explanation is that CS and GWO do not employ any special convergence-enhancement technique compared to the other tested algorithms.

5. CONCLUSIONS

The current paper presented the Distributed Grey Wolf Optimizer (DGWO) algorithm that is based on a distribution model called the island model. The population in DGWO is divided into small populations in an attempt to enhance the diversity of the population and the run-time of the algorithm. DGWO applies the original GWO to the population of each island and then allows selected solutions to be exchanged among the islands based on the random ring topology and the best-worst migration policy.

Different experimental cases were designed and used to study the sensitivity of the performance of DGWO to the parameters of the island model (number of islands *s*, migration frequency M_f and migration rate M_r). The overall experimental results suggest that high values of *s* and low values of M_f significantly improve the performance of DGWO. However, there is no clear indication whether high values or low values of M_r improve the performance of DGWO.

In addition, 30 functions of CEC2014 (real-parameter single-objective optimization-benchmark suit) have been used to compare the performance of DGWO to the performance of well-known optimization algorithms (CS, L-SHADE, MHDA, FWA-DM). The results indicate that DGWO produces competitive results compared to those produced by the other compared algorithms. Interestingly, DGWO produced the lowest FEV for 10 functions of the 30 test functions of CEC2014. This is expected, because DGWO synchronously applies GWO to several islands, which accelerates its convergence to good solutions. Moreover, DGWO provides better chances for unfit candidate solutions in each island to evolve to better candidate solutions.

There are four interesting directions for future work. First, it would be interesting to incorporate the island model to multi-objective discrete GWO [38] to explore its efficiency in solving the scheduling problem in welding production. This scheduling problem is one of the most time-consuming processes in modern industry. Second, a binary version of DGWO will be developed and used to solve the problem of feature selection [35], [68]. Feature selection is normally considered as a complex time-consuming problem when it is used with large datasets. Third, hierarchical Q-learning [69]-[70] and cooperative Q-learning [71]-[72] require heavy and complex computations to efficiently solve learning problems with large state space or action space. Based on the fact that the population of Q-values (i.e., values of state-action pairs in Q-learning) in Q-learning can be represented as an optimization problem [1], [17], [66] and [71], the DGWO algorithm will be applied to hierarchical Q-learning [69]-[70] and cooperative Q-learning [71]-[72] as discussed in [17], [66]. Finally, the experimental results in Section 4.3 demonstrated that L-SHADE [62] performs better than many powerful optimization algorithms. Therefore, the incorporation of the island model into the L-SHADE algorithm will be addressed in a future study in an attempt to elevate the performance of L-SHADE.

ACKNOWLEDGEMENTS

The authors would like to thank Dr. Oguz Emrah Turgut for the source code of GWO that he made publically, [Online], available at:
https://www.researchgate.net/publication/281939423_Grey_Wolf_Optimizer_java_code.

REFERENCES

[1] B. H. Abed-alguni, D. J. Paul, S. K. Chalup and F. A. Henskens, "A Comparison Study of Cooperative Q-learning Algorithms for Independent Learners," Int. J. Artif. Intell., vol. 14, no. 1, pp. 71-93, 2016.

[2] X.-S. Yang, "A New Metaheuristic Bat-inspired Algorithm," Nature Inspired Cooperative Strategies for Optimization (NICSO 2010), Springer, pp. 65-74, 2010.

[3] X.-S. Yang and S. Deb, "Cuckoo Search via Lévy Flights," IEEE World Congress on Nature & Biologically Inspired Computing (NaBIC 2009), pp. 210-214, 2009.

[4] B. H. Abed-alguni and F. Alkhateeb, "Novel Selection Schemes for Cuckoo Search," Arabian Journal for Science and Engineering, vol. 42, no. 8, pp. 3635-3654, 2017.

[5] F. Alkhateeb and B. H. Abed-alguni, "A Hybrid Cuckoo Search and Simulated Annealing Algorithm," Journal of Intelligent Systems, 2017, [Online], Available: https://doi.org/10.1515/jisys-2017-0268.

[6] B. H. Abed-alguni and F. Alkhateeb, "Intelligent Hybrid Cuckoo Search and β-hill Climbing Algorithm," Journal of King Saud University - Computer and Information Sciences, pp. 1-44, 2018, [Online], Available: https://doi.org/10.1016/j.jksuci.2018.05.003.

[7] B. H. Abed-alguni and A. F. Klaib, "Hybrid Whale Optimization and β-hill Climbing Algorithm," International Journal of Computing Science and Mathematics, pp. 1-13, 2018.

[8] S. Mirjalili and A. Lewis, "The Whale Optimization Algorithm," Advances in Engineering Software, vol. 95, pp. 51-67, 2016.

[9] G. Kaur and S. Arora, "Chaotic Whale Optimization Algorithm," Journal of Computational Design and Engineering, vol. 5, no. 3, pp. 275-284 , July 2018.

[10] S. Arora and P. Anand, "Learning Automata Based Butterfly Optimization Algorithm for Engineering Design Problems," International Journal of Computational Materials Science and Engineering, July 2018.

[11] S. Arora and S. Singh, "Butterfly Optimization Algorithm: A Novel Approach for Global Optimization," Soft Computing, pp. 1-20, 2018.

[12] S. Arora and S. Singh, "A Hybrid Optimization Algorithm Based on Butterfly Optimization Algorithm and Differential Evolution," International Journal of Swarm Intelligence, vol. 3, no. 2-3, pp. 152-169, 2017.

[13] S. Arora and S. Singh, "An Improved Butterfly Optimization Algorithm for Global Optimization," Advanced Science, Engineering and Medicine, vol. 8, no. 9, pp. 711-717, 2016.

[14] S. Arora, S. Singh and K. Yetilmezsoy, "A Modified Butterfly Optimization Algorithm for Mechanical Design Optimization Problems," Journal of the Brazilian Society of Mechanical Sciences and Engineering, vol. 40, no. 1, p. 21, 2018.

[15] S. Arora and P. Anand, "Chaotic Grasshopper Optimization Algorithm for Global Optimization," Neural Computing and Applications, pp. 1-21, 2018.

[16] S. Arora and P. Anand, "Chaos-enhanced Flower Pollination Algorithms for Global Optimization," Journal of Intelligent & Fuzzy Systems, vol. 33, no. 6, pp. 3853-3869, 2017.

[17] B. H. Abed-alguni, "Bat Q-learning Algorithm," Jordanian Journal of Computers and Information Technology (JJCIT), vol. 3, no. 1, pp. 56-77, 2017.

[18] S. J. Mousavirad and H. Ebrahimpour-Komleh, "Multilevel Image Thresholding Using Entropy of Histogram and Recently Developed Population-based Metaheuristic Algorithms," Evolutionary Intelligence, vol. 10, no. 1-2, pp. 45-75, 2017.

[19] S. Pare, A. Bhandari, A. Kumar and G. Singh, "Rényi's Entropy and Bat Algorithm Based Color Image Multilevel Thresholding," Machine Intelligence and Signal Analysis: Springer, pp. 71-84, 2019.

[20] R.-E. Precup, R.-C. David, A.-I. Szedlak-Stinean, E. M. Petriu and F. Dragan, "An Easily Understandable Grey Wolf Optimizer and Its Application to Fuzzy Controller Tuning," Algorithms, vol. 10, no. 2, p. 68, 2017.

[21] J. Vaščák, "Adaptation of Fuzzy Cognitive Maps by Migration Algorithms," Kybernetes, vol. 41, no. 3/4, pp. 429-443, 2012.

[22] T. Jayabarathi, T. Raghunathan and A. Gandomi, "The Bat Algorithm, Variants and Some Practical Engineering Applications: A Review," Nature-Inspired Algorithms and Applied Optimization: Springer, pp. 313-330, 2018.

[23] S. K. Sarangi, R. Panda, P. K. Das and A. Abraham, "Design of Optimal High Pass and Band Stop FIR Filters Using Adaptive Cuckoo Search Algorithm," Engineering Applications of Artificial Intelligence, vol. 70, pp. 67-80, 2018.

[24] R.-E. Precup, R.-C. David and E. M. Petriu, "Grey Wolf Optimizer Algorithm-based Tuning of Fuzzy Control Systems with Reduced Parametric Sensitivity," IEEE Transactions on Industrial Electronics, vol. 64, no. 1, pp. 527-534, 2017.

[25] N. Jayakumar, S. Subramanian, S. Ganesan and E. Elanchezhian, "Grey Wolf Optimization for Combined Heat and Power Dispatch with Cogeneration Systems," International Journal of Electrical Power & Energy Systems, vol. 74, pp. 252-264, 2016.

[26] M. Nouiri, A. Bekrar, A. Jemai, S. Niar and A. C. Ammari, "An Effective and Distributed Particle Swarm Optimization Algorithm for Flexible Job-Shop Scheduling Problem," Journal of Intelligent Manufacturing, vol. 29, no. 3, pp. 603-615, 2018.

[27] M. K. Marichelvam and M. Geetha, "Cuckoo Search Algorithm for Solving Real Industrial Multi-Objective Scheduling Problems," Encyclopedia of Information Science and Technology, 4th Edition: IGI Global, pp. 4369-4381, 2018.

[28] S. Mirjalili, S. M. Mirjalili and A. Lewis, "Grey Wolf Optimizer," Advances in Engineering Software, vol. 69, pp. 46-61, 2014.

[29] M. A. Tawhid and A. F. Ali, "A Hybrid Grey Wolf Optimizer and Genetic Algorithm for Minimizing Potential Energy Function," Memetic Computing, vol. 9, no. 4, pp. 347-359, 2017.

[30] W. Gai, C. Qu, J. Liu and J. Zhang, "An Improved Grey Wolf Algorithm for Global Optimization," 2018Chinese Control and Decision Conference (CCDC), pp. 2494-2498, 2018.

[31] M. A. Al-Betar and M. A. Awadallah, "Island Bat Algorithm for Optimization," Expert Systems with Applications, vol. 107, pp. 126-145, 2018.

[32] M. A. Al-Betar, M. A. Awadallah, A. T. Khader and Z. A. Abdalkareem, "Island-based Harmony Search for Optimization Problems," Expert Systems with Applications, vol. 42, no. 4, pp. 2026-2035, 2015.

[33] A. L. Corcoran and R. L. Wainwright, "A Parallel Island Model Genetic Algorithm for the Multiprocessor Scheduling Problem," Proceedings of the 1994 ACM Symposium on Applied Computing, pp. 483-487, 1994.

[34] E. Emary and H. M. Zawbaa, "Impact of Chaos Functions on Modern Swarm Optimizers," PLOS One, vol. 11, no. 7, p. e0158738, 2016.

[35] E. Emary, H. M. Zawbaa and A. E. Hassanien, "Binary Grey Wolf Optimization Approaches for Feature Selection," Neurocomputing, vol. 172, pp. 371-381, 2016.

[36] T. Jayabarathi, T. Raghunathan, B. R. Adarsh and P. N. Suganthan, "Economic Dispatch Using Hybrid Grey Wolf Optimizer," Energy, vol. 111, pp. 630-641, 2016.

[37] M. Pradhan, P. K. Roy and T. Pal, "Grey Wolf Optimization Applied to Economic Load Dispatch Problems," International Journal of Electrical Power & Energy Systems, vol. 83, pp. 325-334, 2016.

[38] C. Lu, S. Xiao, X. Li and L. Gao, "An Effective Multi-objective Discrete Grey Wolf Optimizer for a Real-world Scheduling Problem in Welding Production," Advances in Engineering Software, vol. 99, pp. 161-176, 2016.

[39] G. Komaki and V. Kayvanfar, "Grey Wolf Optimizer Algorithm for the Two-stage Assembly Flow Shop Scheduling Problem with Release Time," Journal of Computational Science, vol. 8, pp. 109-120, 2015.

[40] M. Ruciński, D. Izzo and F. Biscani, "On the Impact of the Migration Topology on the Island Model," Parallel Computing, vol. 36, no. 10, pp. 555-571, 2010.

[41] M. Tomassini, Spatially Structured Evolutionary Algorithms: Artificial Evolution in Space and Time, Springer, 2006.

[42] M. Tomassini, "Spatially Structured Evolutionary Algorithms: Artificial Evolution in Space and Time (Natural Computing Series), Secaucus," Ed: NJ, USA: Springer-Verlag New York, Inc, 2005.

[43] D. Jitkongchuen, "A Hybrid Differential Evolution with Grey Wolf Optimizer for Continuous Global Optimization," IEEE 7th International Conference on Information Technology and Electrical Engineering (ICITEE), pp. 51-54, 2015.

[44] H. M. Zawbaa, E. Emary, C. Grosan and V. Snasel, "Large-dimensionality Small-instance Set Feature Selection: A Hybrid Bio-inspired Heuristic Approach," Swarm and Evolutionary Computation, vol. 42, pp. 29-42, 2018.

[45] S. Saremi, S. Z. Mirjalili and S. M. Mirjalili, "Evolutionary Population Dynamics and Grey Wolf Optimizer," Neural Computing and Applications, vol. 26, no. 5, pp. 1257-1263, 2015.

[46] L. Rodríguez, O. Castillo and J. Soria, "A Study of Parameters of the Grey Wolf Optimizer Algorithm for Dynamic Adaptation with Fuzzy Logic," Nature-Inspired Design of Hybrid Intelligent Systems: Springer, pp. 371-390, 2017.

[47] H. Joshi and S. Arora, "Enhanced Grey Wolf Optimization Algorithm for Constrained Optimization Problems," International Journal of Swarm Intelligence, vol. 3, no. 2-3, pp. 126-151, 2017.

[48] M. R. S. Malik, E. R. Mohideen and L. Ali, "Weighted Distance Grey Wolf Optimizer for Global Optimization Problems," IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), pp. 1-6, 2015.

[49] E. A. Emary, H. M. A. Zawbaa and C. A. Grosan, "Experienced Grey Wolf Optimizer through Reinforcement Learning and Neural Networks," vol. 29, no. 3, pp. 681-694, 2018.

[50] H. Joshi and S. Arora, "Enhanced Grey Wolf Optimization Algorithm for Global Optimization," Fundamenta Informaticae, vol. 153, no. 3, pp. 235-264, 2017.

[51] M. Kohli and S. Arora, "Chaotic Grey Wolf Optimization Algorithm for Constrained Optimization Problems," Journal of Computational Design and Engineering, vol. 5, no. 4, pp. 458-472,2017.

[52] A. A. Heidari and P. Pahlavani, "An Efficient Modified Grey Wolf Optimizer with Lévy Flight for Optimization Tasks," Applied Soft Computing, vol. 60, pp. 115-134, 2017.

[53] M. A. Al-Betar, I. A. Doush, A. T. Khader and M. A. Awadallah, "Novel Selection Schemes for Harmony Search," Applied Mathematics and Computation, vol. 218, no. 10, pp. 6095-6117, 2012.

[54] S. Gupta and K. Deep, "A Novel Random Walk Grey Wolf Optimizer," Swarm and Evolutionary Computation, 2018.

[55] S. Gupta and K. Deep, "Random Walk Grey Wolf Optimizer for Constrained Engineering Optimization Problems," Computational Intelligence, 2018.

[56] Fr et al., "A Dynamic Island-based Genetic AlgorithmsFramework," Proceedings of the 8th International Conference on Simulated Evolution and Learning, Kanpur, India, 2010.

[57] H. T. T. Thein, "Island Model Based Differential Evolution Algorithm for Neural Network Training," Advances in Computer Science: An International Journal (ACSIJ), vol. 3, no. 1, pp. 67-73, 2014.

[58] Z. A. Mostafa, N. H. Awad and R. M. Duwairi, "Multi-objective Differential Evolution Algorithm with A New Improved Mutation Strategy," International Journal of Artificial Intelligence^{TM}, vol. 14, no. 2, pp. 23-41, 2016.

[59] J. F. Romero and C. Cotta, "Optimization by Island-structured Decentralized Particle Swarms," Computational Intelligence, Theory and Applications: Springer, pp. 25-33, 2005.

[60] M. Randall and A. Lewis, "A Parallel Implementation of Ant Colony Optimization," Journal of Parallel and Distributed Computing, vol. 62, no. 9, pp. 1421-1432, 2002.

[61] S. Gupta and K. Deep, "Performance of Grey Wolf Optimizer on Large Scale Problems," AIP Conference Proceedings, vol. 1802, no. 1, p. 020005, 2017, AIP Publishing.

[62] R. Tanabe and A. S. Fukunaga, "Improving the Search Performance of SHADE Using Linear Population Size Reduction," IEEE Congress on Evolutionary Computation (CEC), pp. 1658-1665, 2014.

[63] K. S.SreeRanjini and S. Murugan, "Memory-based Hybrid Dragonfly Algorithm for Numerical Optimization Problems," Expert Systems with Applications, vol. 83, pp. 63-78, 2017.

[64] C. Yu, L. Kelley, S. Zheng and Y. Tan, "Fireworks Algorithm with Differential Mutation for Solving the CEC 2014 Competition Problems," IEEE Congress on Evolutionary Computation (CEC), pp. 3238-3245, 2014.

[65] P. N. Suganthan et al., "Problem Definitions and Evaluation Criteria for the CEC 2005 Special Session on Real-parameter Optimization," KanGAL Report, vol. 2005005, p. 2005, 2005.

[66] B. H. Abed-alguni, "Action-Selection Method for Reinforcement Learning Based on Cuckoo Search Algorithm," Arabian Journal for Science and Engineering, pp. 1-15, 2017.

[67] J. Liang, B. Qu and P. Suganthan, "Problem Definitions and Evaluation Criteria for the CEC 2014 Special Session and Competition on Single Objective Real-parameter Numerical Optimization," Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou, China; and Technical Report, Nanyang Technological University, Singapore, 2013.

[68] E. Emary, H. M. Zawbaa, C. Grosan and A. E. Hassenian, "Feature Subset Selection Approach by Gray-wolf Optimization," Afro-European Conference for Industrial Advancement, pp. 1-13, Springer, 2015.

[69] B. H. Abed-alguni, S. K. Chalup, F. A. Henskens and D. J. Paul, "A Multi-agent Cooperative Reinforcement Learning Model Using a Hierarchy of Consultants, Tutors and Workers," Vietnam Journal of Computer Science, vol. 2, no. 4, pp. 213-226, 2015.

[70] B. H. Abed-alguni, S. K. Chalup, F. A. Henskens and D. J. Paul, "Erratum to: A Multi-agent Cooperative Reinforcement Learning Model Using a Hierarchy of Consultants, Tutors and Workers," Vietnam Journal of Computer Science, vol. 2, no. 4, pp. 227-227, 2015.

[71] B. H. K. Abed-alguni, "Cooperative Reinforcement Learning for Independent Learners," 2014.

[72] B. H. Abed-alguni and M. A. Ottom, "Double Delayed Q-learning," International Journal of Artificial Intelligence, vol. 16, no. 2, pp. 41-59, 2018.

ملخص البحث:

تعد خوارزمية "الذئاب الرمادية" من الخوارزميات المهمة المستخدمة في الأمثلة والمبنية على تقنية السِّرب. وقد استوحيت من استراتيجية المطاردة وهرمية القيادة التي تتبعها الذئاب الرمادية. وقد استخدمت بنجاح لحل مشكلات متعددة من المشكلات المستمرة والمجرّدة لعملية الأمثَلة. ومع ذلك، فإن السلبية الرئيسية لها تكمن في أنها قد تتجمع في شكل حلول شبه مثالية في مراحل مبكرة من عملية المحاكاة بسبب فقدان التنوع في مجتمع الخوارزمية.

هذه الورقة تقدم تغييراً موزعاً لخوارزمية الذئاب الرمادية في محاولة لتحسين التنوع في الخوارزمية، وذلك عبر تنظيم مجتمع الخوارزمية في هيئة مجموعات مستقلة صغيرة (جُزُر) بناءً على النموذج المعروف بنموذج الجُزُر. الخوارزمية المقترحة تطبق الخوارزمية الأصلية على كل جزيرة من الجُزُر، مما يسمح بتبادل الحلول المختارة بين الجُزر بناءً على تقنية الحلقات العشوائية وسياسة الهجرة بين الأفضل والأسوأ. وتوفر الخوارزمية المعدّلة بيئة أفضل للحلول المرشحة غير المهيأة في كل جزيرة كي تتطور الى حلولٍ أفضل، الأمر الذي يزيد من احتمالية إيجاد الحلول المثالية.

ومن السِّمات المهمة الأخرى للخوارزمية المعدّلة المقترحة أنها يمكن أن تعمل باستخدام أجهزة متوازية، الأمر الذي يعني إمكانية التقليل من تعقيد الحسابات مقارنة بالصيغ القائمة من خوارزمية الذئاب الرمادية. ومن ناحية أخرى، تمّ تقييم الخوارزمية المقترحة في هذه الورقة من خلال مقارنتها بأنواع أخرى من الخوارزميات المبنية على تقنية السِّرب. كذلك جرى تقييم حساسية الخوازمية المقترحة لمتغيراتها باستخدام خمس عشرة دالّة اختبار معيارية. وأشارت المقارنة واختبار الحساسية المشار إليهما الى أن الخوارزمية المقترحة تؤدي أداءً منافساً بالمقارنة مع الخوارزميات الأخرى.

ENHANCED ULTRA-WIDE BAND HEXAGONAL PATCH ANTENNA

Yanal S. Al-Faouri, Noor M. Awad and Mohamed K. Abdelazeez

(Received: 14-Aug.-2018, Revised: 17-Sep.-2018, Accepted: 24-Sep.-2018)

ABSTRACT

An enhanced hexagonal shaped planar antenna is presented for ultra-wideband (UWB) applications. In this paper, a hexagonal patch with six circular cuts at its vertices is designed on FR4-substrate with 50 Ω microstrip triangular tapered feed line and a bevelled partial ground plane with five half circular sleeves. The design is investigated using the high-frequency structure simulator (HFSS). The simulated and measured scattering parameter S_{11} (Reflection Coefficient) results show good impedance matching in the frequency range (3 - 27.57 GHz) satisfying return loss (RL = $|S_{11}|$) ≥ 10 dB with a percentage bandwidth (PBW) of 160.75%. High gain and efficiency, radiation pattern similar to the electric dipole in E-plane and good omnidirectionality in the H-plane are achieved.

KEYWORDS

Ultra-Wide Band (UWB), Reflection coefficient, Circular cuts, Sleeves, Bevel, Gain and bandwidth.

1. INTRODUCTION

The most important requirements in modern communication systems are to provide a wide frequency range with very low power consumption. The UWB wireless technology is launched in 2002 by the Federal Communication Commission (FCC) which authorizes the unlicensed use of the frequency band 3.1 to 10.6 GHz with PBW ($\text{PBW} = 100\% * \text{bandwidth}/\text{center frequency}$) of 91.33% and -41.3 dBm/MHz maximally allowed radiated power [1]. Nowadays, the new wireless communication systems in military and civilian applications are searching for an enhanced wideband that covers both the short and long frequency ranges.

Different techniques have been proposed by different researchers to enhance the antenna bandwidth with a low profile and compact size. Rectangular patch with one round cut in its four corners with one ground groove that has a shape composed of triangle and rectangle to get 8.28 GHz bandwidth (3.42 - 11.7 GHz) with a PBW equal to 109.52% is presented in [2]. Three ground plane modifications consisting of two rectangular sleeves, two rectangular slots and one rectangular groove are introduced in [3] to achieve a bandwidth of 19 GHz (3.4 - 22.4 GHz) with a PBW equal to 147.29%. Two trapezoidal patches are etched on both sides of the substrate with a microstrip feed line to increase the PBW to 114.28% [4]. Adding three steps in the lower patch corners of rectangular shape patch and using microstrip feed line are proposed in [5] to increase the bandwidth to (2.33 - 12.4 GHz) with a PBW equal to 136.73%. Adding a rectangular slit and attaching L- and T- shaped stubs on the radiating circular patch with an offset feed achieve a PBW of 127.87% (3.08 to over 14 GHz) [6]. Cutting a bevel in the rectangular patch and etching two rounded inverted L-shaped slots with an open end in the square ground plane achieve a PBW of 129.18% (2.7 - 12.55 GHz) [7]. A triangular patch with one rectangular slot and two slits fed by coplanar feed line with the defected ground are used to increase the PBW to 112.5% (2.8 - 10 GHz) [8].

Hexagonal patch antennas are studied by different researchers and achieve good PBW. Folded hexagon UWB patch antenna with an offset feed at one of its vertices achieves a PBW of 144.33% (2.796 - 17.296 GHz) [9]. Hexagonal patch antenna with two symmetrical slots is etched at the center of the patch with a microstrip feed line to achieve a PBW of 109.09% (2 - 6.8 GHz) [10]. A spanner shape hexagonal patch antenna is designed by defecting the patch with a rectangular shape slot to achieve a PBW of 118.79% (2.95 - 11.58 GHz) [11]. A coplanar waveguide (CPW)-fed hexagonal patch antenna

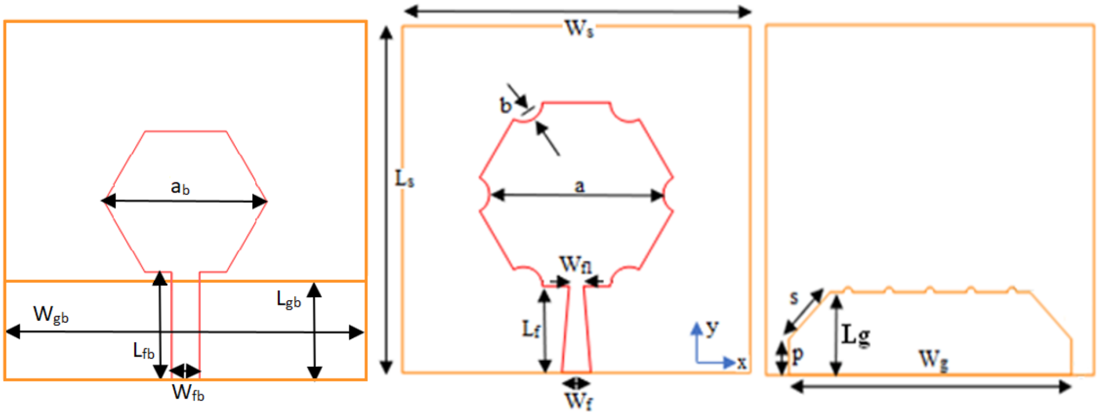
Y. S. Al-Faouri[1], N. M. Awad[2] and M. K. Abdelazeez[3] are with Electrical Engineering Department, The University of Jordan, Amman, Jordan.
Emails: [1]y.faouri@ju.edu.jo, [2]n.awad@ju.edu.jo and [3]abdelazeez@ieee.org.

with six small hexagonal elements (fractal elements) is added to its corners to achieve a PBW of 93.33% (4 - 11 GHz) [12]. Different configurations of hexagonal shape patch antennas are introduced in [13] with defected ground planes to reduce the antenna size without affecting the bandwidth. One is composed of the hexagonal patch with L-shape and bevel slots and four rectangular slots in the ground plane to achieve a PBW of 120.43% (2.98 - 12 GHz). While the other consists of the hexagonal patch with one horizontal rectangular slot and bevel slots, one circular slot on the feed line and four rectangular slots in the ground plane to achieve a PBW of 135.2% (2.9 - 15 GHz). A new hexagonal patch antenna is proposed in [14] which consists of small five trapezoidal elements which are added to the center of the patch edges, small six hexagonal slots on each of its corners and another hexagonal slot added at the patch center to achieve a PBW of 126.06% (3.1 - 13.67 GHz). Two hexagonal patch antennas are proposed in [15], one of them achieves the UWB with PBW of 129.55% (3.1 - 14.5 GHz) by etching a rectangular groove and cross slot in the ground plane. The other is designed for the super-wideband antenna (SWB) to achieve a PBW of 154.61% (3.2 - 25 GHz) by adding a deep groove that divides the ground plane into two strips laid symmetrically around the feed and triangular slots at the upper corners of the ground plane.

In this paper, a new enhanced hexagonal UWB microstrip antenna design is proposed and investigated. The antenna shape and dimensions are outlined in Section 2. The proposed antenna consists of a triangular tapered microstrip feed line, a hexagonal radiation patch with six circular cuts at the patch vertices and a bevelled partial ground plane with five half circular sleeves. The simulation results and discussions are presented in Section 3. The experimental verifications are outlined in Section 4. Finally, the conclusion is given in Section 5.

2. Antenna Structure

The proposed antenna with the geometrical parameters is shown in Figure 1, where all dimensions are obtained carefully by parametric analysis (explained in Section 3) in order to achieve the desired bandwidth over the needed frequency range. The antenna dimensions (in mm) are: the substrate is FR4 - epoxy with a thickness $h = 1.6$, $\tan\delta = 0.02$, $\varepsilon_r = 4.4$, width $W_S = 36$ and length $L_S = 36$. A triangular tapered feed line is designed using the equations given in [16] and the length of the triangular tapered feed line is chosen approximately equal to the guided wavelength with $L_f = 9$, $W_f = 3$ and $W_{f1} = 1.5$. The distance between the hexagonal patch and the circular cut edges at the patch middle is $a = 18$ and six circular cuts at the vertices with a radius $b = 2$. The partial ground plane length $L_g = 8$ and width $W_g = 31$. The ground bevel has a length $s = 7$ and is located at a distance $p = 3.5$ from the lower substrate edge. In order to enhance the design, five half circular sleeves are attached to the ground plane in the bottom layer with a radius $r_s = 0.5$.



(a) (b) (c)

Figure 1. The hexagonal antenna structure; (a) the basic model, (b) the proposed antenna top layer and (c) the proposed antenna bottom layer.

3. Discussion and Parametric Study

The design started with the basic model which consists of a simple hexagonal patch fed by a rectangular microstrip feed line with a partial ground plane as shown in Figure 1.

(a) (where $L_{gb} = 11$ mm, $W_{gb} = 40$ mm, $L_{fb} = 12$ mm, $W_{fb} = 3$ mm and $a_b = 18$ mm). The achieved PBW for this design is very low with bad impedance matching. The enhancement of the bandwidth is achieved using different means, including changing the length and width of the ground plane, adding sleeves to the partial ground plane, changing the length of the tilted ground plane edge, adding patch circular cuts and adjusting the microstrip feed line shape. The effect of each parameter on the bandwidth (with the scattering parameter $S_{11} \leq$ *- 10 dB)* was performed and optimized to reach the final design. The parametric results in the paper are generated from the final design by varying one parameter at a time and keeping all other parameters constant as listed in Section 2 to show the effect of each parameter alone. The bandwidth enhancement process is as follows:

3.1 Ground Plane Modifications

The ground plane dimensions are important parameters in the design of the UWB antennas since the bandwidth depends strongly on them [17]. The simulation results for the scattering parameters S_{11}using HFSS simulator for $L_g$equal to 7, 8 and 9 mm are shown in Figure 2. High values of L_g cause narrowband with a lot of rejection bands, while decreasing L_g leads to high bandwidth with lower starting operating frequency. Choosing L_g = 8 mm achieves higher bandwidth, but it still needs to adjust the impedance matching over the different frequency ranges.

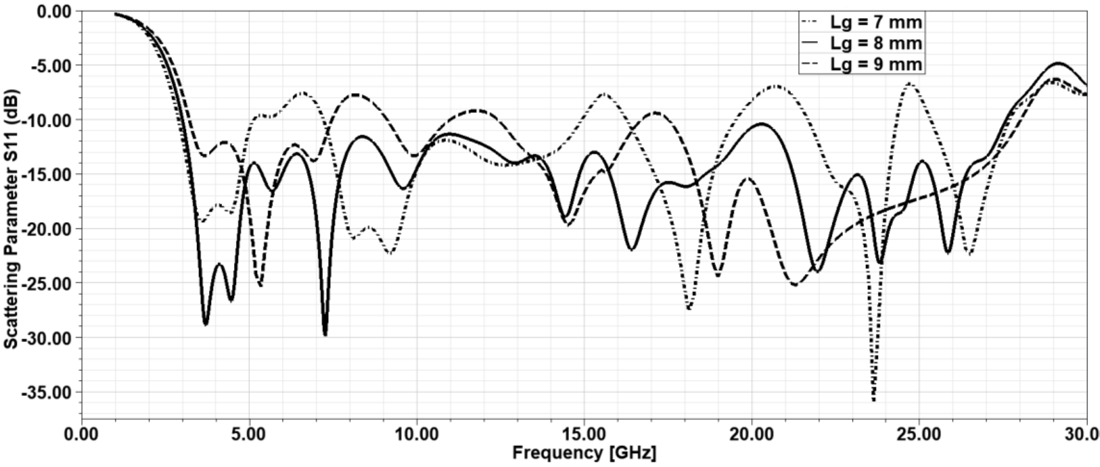


Figure 2. The reflection coefficient when varying the ground length (L_g).

A parametric study is also conducted on the ground plane width (W_g) between (29 - 33) mm, where better impedance matching is achieved with lower starting operating frequency when W_g = 31 mm. The presence of sleeves in the ground plane increases the inductive part of the input impedance and generates additional resonant mode to improve the overall bandwidth [3, 18]. Parametric analysis is performed on the half circular sleeves' parameters to investigate their influence on the antenna performance. The sleeve parameters are the sleeve numbers (N) and their radii (r_s), where r_s is varied between (0 - 1) mm, while N is between (3 - 7). The simulated reflection coefficient for the sleeve number (N) variation shows that using N = 5 achieves lower starting operating point, wider bandwidth and better impedance matching. The reflection coefficient for the sleeve radius variations is shown in Figure 3, where it can be noticed that varying $r_s$will affect mainly the impedance matching and consequently the covered bandwidth. The optimum value to use for the proposed antenna is r_s = 0.5 mm and N = 5.

Bevelling the ground top corners with symmetrical tilted cuts generate more resonant frequencies and adjust the input impedance imaginary part which leads to wider impedance bandwidth [19]-[20]. Parametric analysis is conducted on the ground cut length (s) between (5 - 9) mm, where the optimum value for s is 7 mm, since below and beyond this value, the impedance matching degrades at different frequency bands.

3.2 Using Triangular Tapered Feed Line

The use of a triangular tapered feed line will enhance the matching between the feeding point and the hexagonal patch to smooth the current path and reduce the incident wave reflection, which achieves wider bandwidth [16]. The reflection coefficient comparison when using rectangular and triangular tapered feed lines and by keeping all dimensions as in Section 2 is shown in Figure 4. It can be observed that higher bandwidth and better impedance matching are obtained when using a triangular tapered feed line.

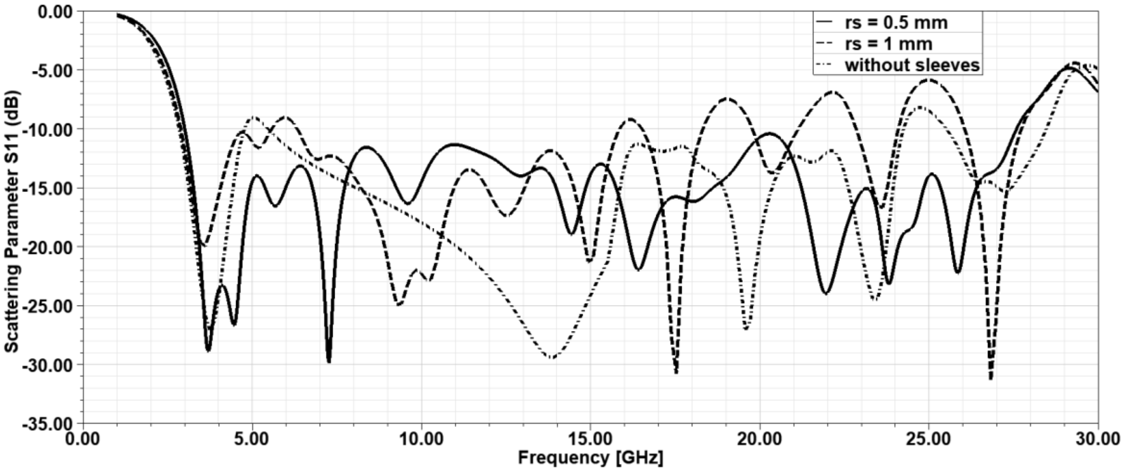


Figure 3. The reflection coefficient when varying the sleeves radius (r_s) using five sleeves.

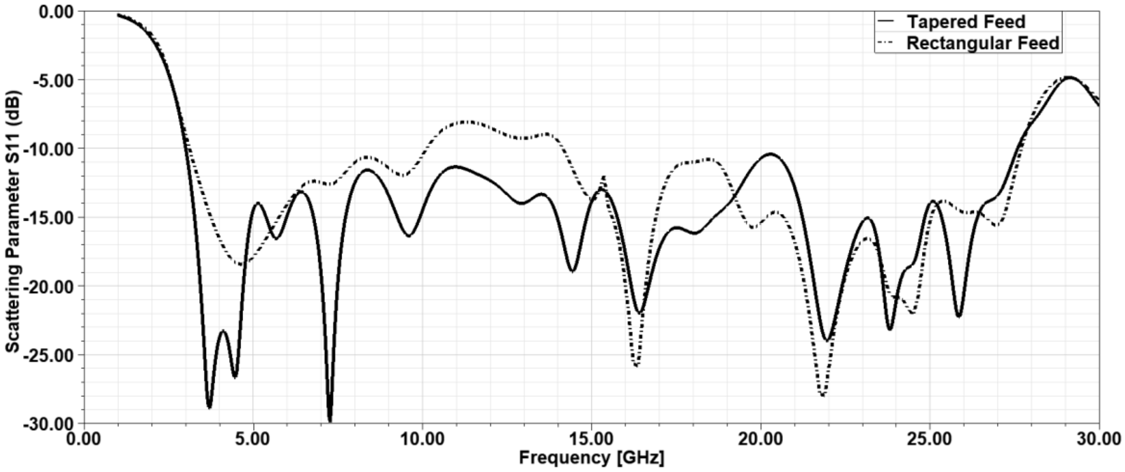


Figure 4. The reflection coefficient to compare between the rectangular and triangular tapered feed lines.

3.3 Incorporating Symmetrical Circular Cuts at the Hexagonal Patch Vertices

Adding cuts in the hexagonal patch leads to perturb the surface current path length which will generate more than one resonant frequency and increase the bandwidth. Parametric analysis is conducted to examine the effect of the circular cuts' parameters on the bandwidth. The important parameters to study are the number of the circular cuts (M) and their radius (b), where M is varied between (0 - 6) and b between (1 - 3) mm. The location of the cuts is varied such as for M = 2 the cuts are on the middle corners and for M = 4 the cuts are in the upper and the lower corners, while for M = 6 the cuts are on every corner. The simulated results for varying the cuts' number M are shown in Figure 5, which confirms that increasing the number of cuts will enhance the impedance matching and the covered bandwidth. The simulated reflection coefficient for radius variation is shown in Figure 6. Varying the cut radius b, the lower operating frequency changes, because the current path on the patch is perturbed. The main noticeable effect of the patch cuts' radius b is at the impedance matching. It is concluded that the optimum value for M=6 and for b = 2 mm. The proposed antenna reflection coefficient *versus* frequency using HFSS software tool before and after all modifications is compared in Figure 7. The achieved bandwidth when RL ≥ 10 dB ranges from 3 till 27.57 GHz with a PBW equal to 160.75% with better impedance matching.

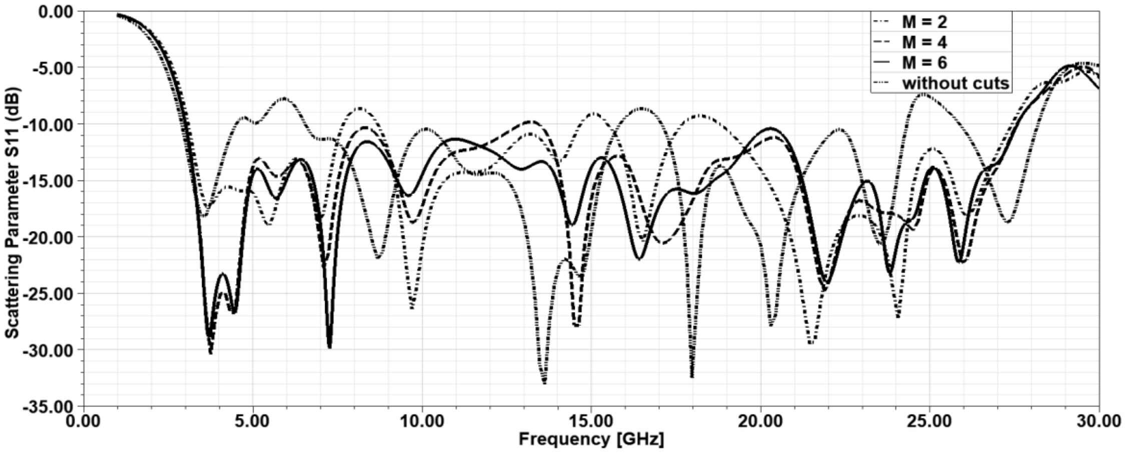


Figure 5. The reflection coefficient when varying the patch circular cut number (M).

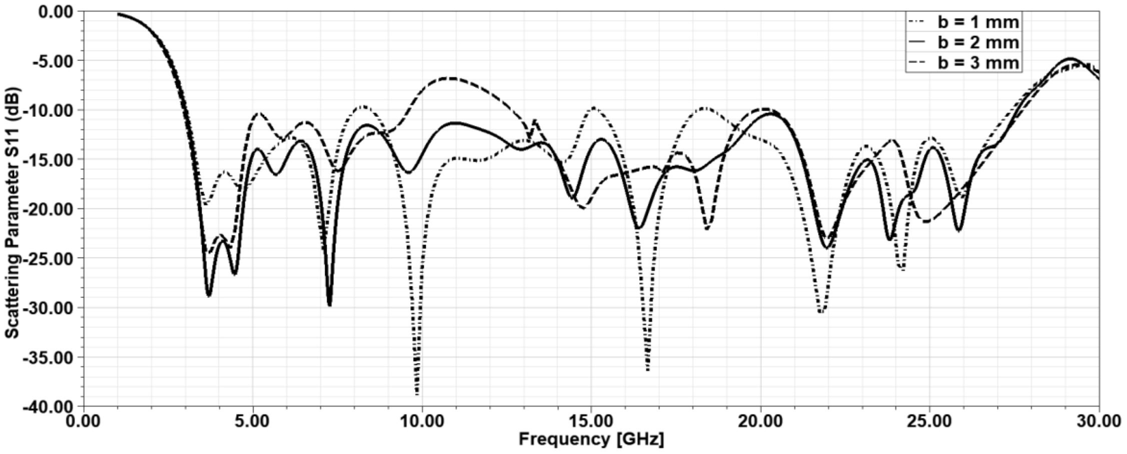


Figure 6. The reflection coefficient when varying the circular patch cut radius (b).

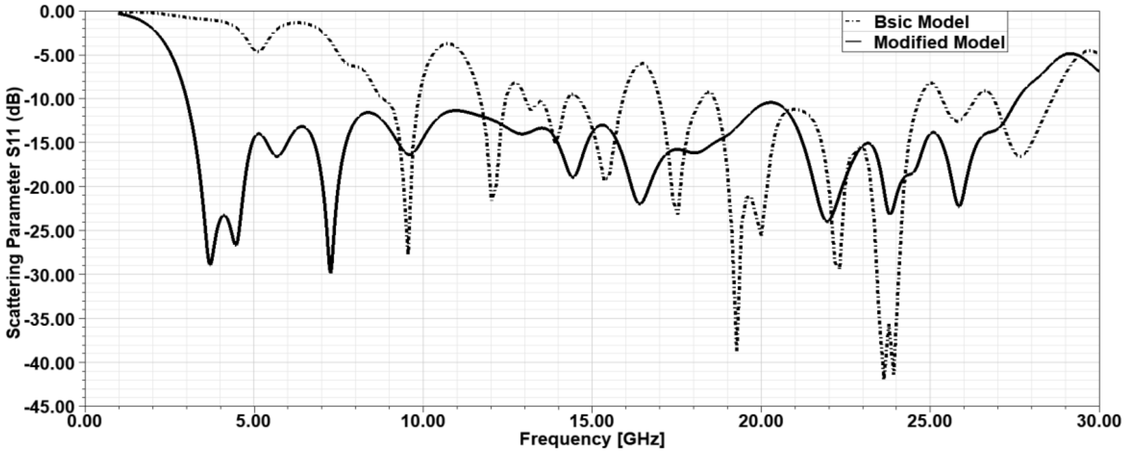


Figure 7. The simulated reflection coefficient for the antenna before and after modifications.

The simulated radiation patterns for the E and H planes at various frequencies 4, 6, 15 and 20 GHz are shown in Figure 8, where the E and H planes are the yz- plane (φ = 90˚ and 0˚ < θ < 180˚) and the xz-plane (φ = 0˚ and 0˚ < θ < 180˚), respectively. In the E-plane, the antenna exhibits a dipole shape at low-frequency range, but because of the existence of higher-order modes, the number of lobes rises with the increase of frequency. The H-plane shows good omnidirectionality at low-frequency range and becomes less omnidirectional with an increase in frequency. Figure 9 shows the simulated peak realized gain, where the gain has a low value at the start of the desired band and increases as a function of the frequency band and ranges between 5 and greater than 7 dB. The radiation efficiency for the proposed antenna is shown in Figure 10, where it starts at98% and ends with 80%.

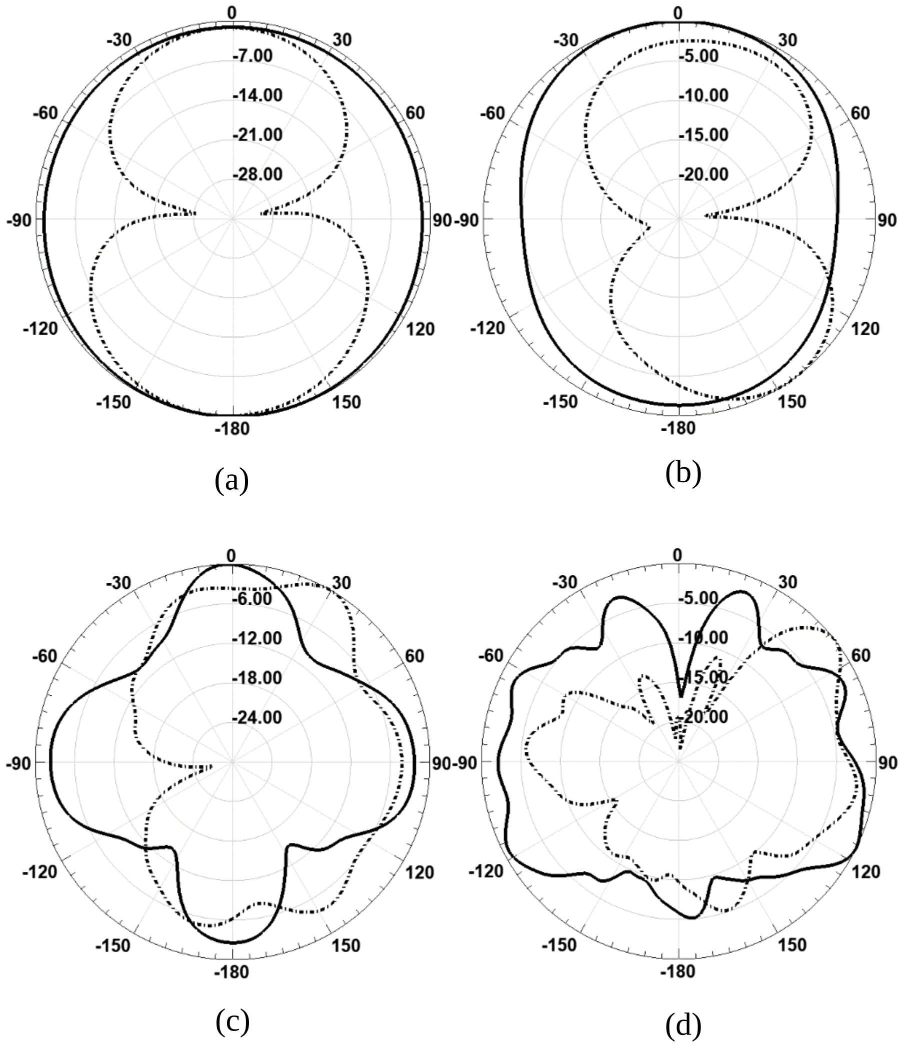


Figure 8. The radiation patterns at:(a) 4 GHz, (b) 6 GHz, (c) 15 GHz and(d) 20 GHz (E--- &H __).

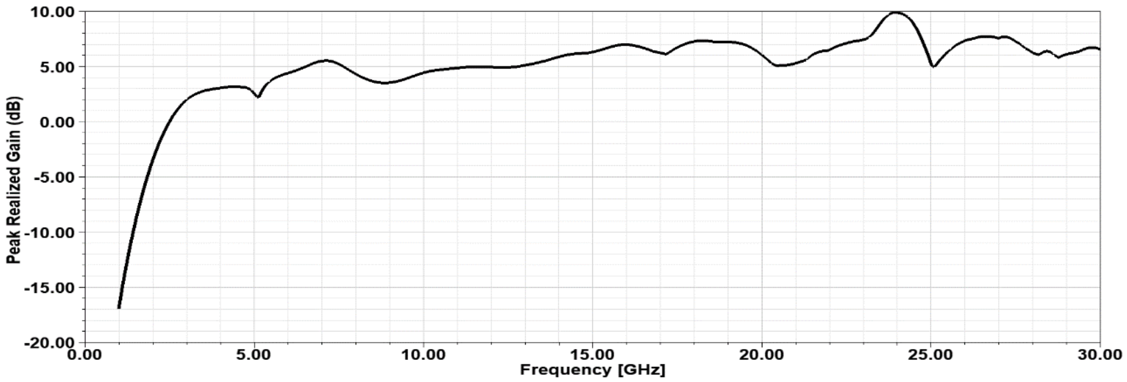


Figure 9. The proposed antenna peak realized gain.

Comparison between different published works and the proposed antenna are shown in Table 1. The proposed antenna achieves a wide impedance bandwidth with simple design compared to other works in [9]-[15]. Although the proposed antenna has a larger substrate area than those in [9], [12]-[13] and [15], the achieved impedance bandwidth is larger.

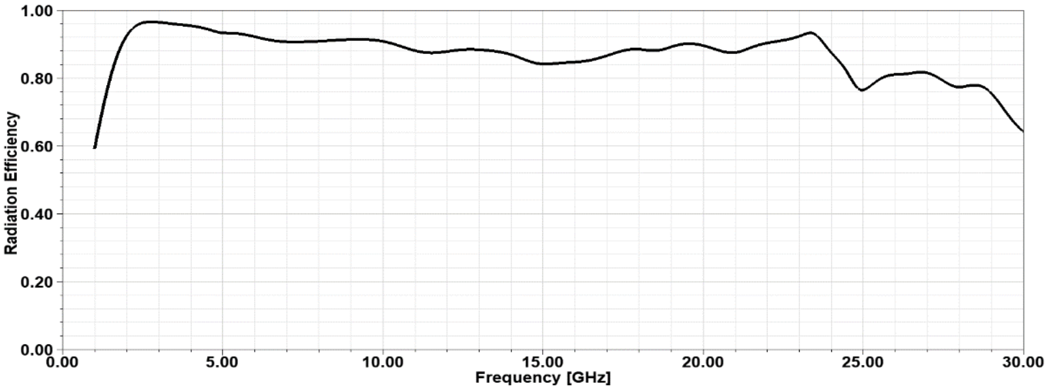


Figure 10. The proposed antenna radiation efficiency.

Table 1. Comparison between the proposed antenna and antennas presented in other published works.

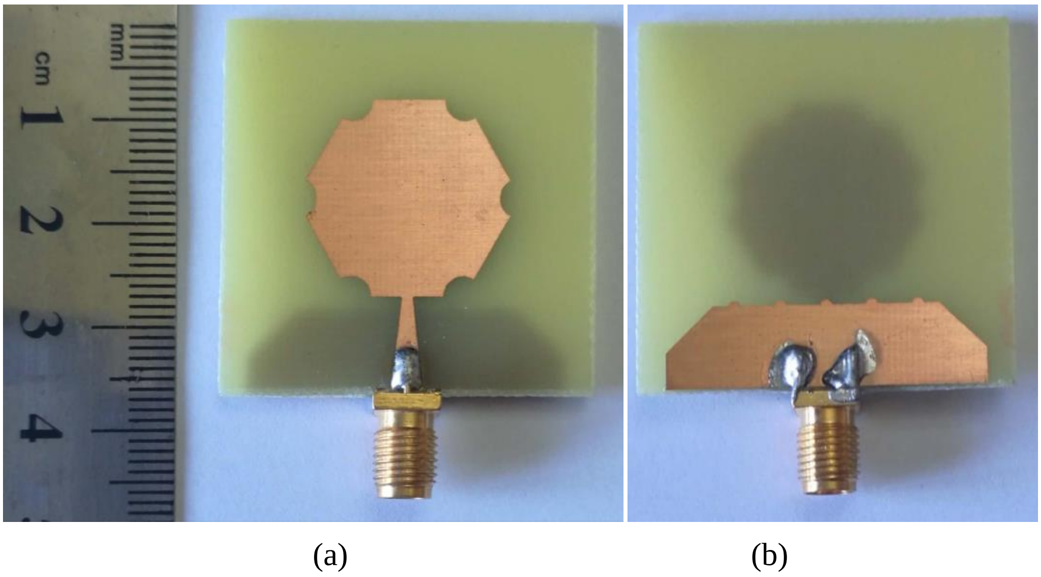
Reference / Parameter	Antenna substrate area (mm^2)	Impedance bandwidth (GHz)	Percentage Bandwidth (PB)
[9]	27 x 24	2.8 - 17.3	144.33 %
[10]	54 x 49	2 - 6.8	109.09%
[11]	31 x 52	2.95 - 11.58	118.79%
[12]	25 x 25	4 - 11	93.33%
[13]	13 x 46.5	2.9 - 15	135.2%
[14]	39 x 36.5	3.1 - 13.67	126.06%
[15]	35.5 x 30.35	(UWB) 3.1 – 14.5 (SWB) 3.2 - 25	(UWB) 129.55% (SWB) 154.61%
Proposed Antenna	**36 x 36**	**3 - 27.57**	**160.57**%

4. Experimental Verification

The designed antenna is fabricated on an FR4 substrate with a dielectric constant ε_r = 4.4 and a height h = 1.6 mm as shown in Figure 11. The facilities at King Abdullah Design and Development Bureau (KADDB) were utilized to test the fabricated antenna. The reflection coefficient is measured using Agilent N5242A vector network analyzer. The simulated and measured reflection coefficient results for the proposed antenna are shown in Figure 12 and they compare favourably. The difference between the measured and the simulated results are due to different factors which are not considered through the simulation process, such as the accuracy and precision of fabrication techniques used, the SMA connector welding and its effect in the frequency range beyond 18 GHz, as well as the non-homogeneous behaviour of the FR4 substrate with frequency variations.

5. Conclusion

A new planar enhanced UWB antenna is designed for UWB applications. The proposed antenna consists of a hexagonal radiation patch with six circular cuts at its vertices, a triangular tapered microstrip feed line and a bevelled partial ground plane with the addition of five half circular sleeves. The design is investigated using electromagnetic simulators HFSS. The simulation results show good impedance matching over (3 - 27.57) GHz for a return loss (RL) ≥ 10 dB. Results of the measurements and simulation compare favourably. Higher gain and efficiency, as well as dipole shape radiation patterns in the E-plane and omni directionality in the H-plane, are obtained.



(a) (b)

Figure 11. The fabricated proposed antenna: (a) patch antenna and (b) ground plane.

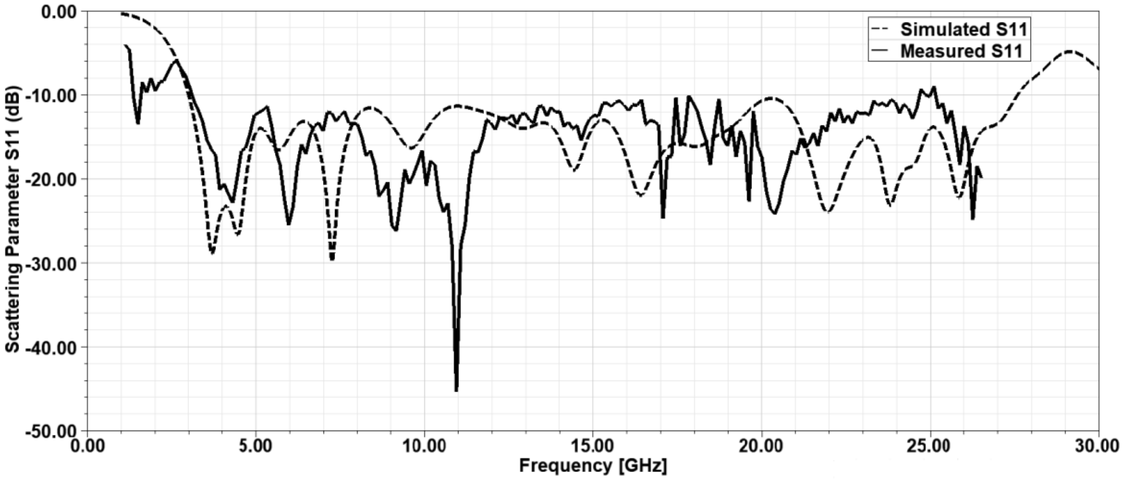


Figure 12. The simulated and measured reflection coefficient curves for the proposed antenna.

ACKNOWLEDGEMENTS

The authors acknowledge the valuable support of the technical staff in the KADDB through the performed measurements of the proposed antenna.

REFERENCES

[1] ET Docket 98 - 153, Federal Communication Commissions (FCC), "First Report and Order, Revision of Part15 of the Commission’s Rules Regarding Ultra-Wideband Transmission Systems," Washington, DC, Technical Report, February 14, 2002.

[2] N. M. Awad and M. K. Abdelazeez, "Multi Slot Microstrip Antenna for Ultra-Wideband Applications," Journal of King Saud University – Engineering Sciences, vol. 30, no. 1, pp. 38-45, 2018.

[3] N. M. Awad, M. K. Abdelazeez and A. Al-Sharif, "Enhanced UWB Printed Monopole Antenna Based on Ground Plane Modifications," Jordanian Journal of Computers and Information Technology (JJCIT), vol. 4, no. 1, 2018.

[4] Y.S. Hu, M. Li, G. P. Gao, J. S. Zhang and M. K. Yang, "A Double-printed Trapezoidal Patch Dipole Antenna for UWB Applications with Band-notched Characteristics," Progress in Electromagnetics Research (PIER), vol. 103, pp. 259-269, 2010.

[5] Z. Ul Abedin and Z. Ullah, "Design of a Microstrip Patch Antenna with High Bandwidth and High Gain for UWB and Different Wireless Applications," International Journal of Advanced Computer Science and Applications (IJACSA), vol. 8, no. 10, 2017.

[6] V. Sharbati, P. Rezaei and M. M. Fakharian, "Compact Planar UWB Antenna with Enhanced Bandwidth and Switchable Band-Notch Function for WLAN and DSRC," IETE Journal of Research, vol. 63, no. 6, pp. 805-812, 2017.

[7] A. Dastranjand F. Bahmanzadeh, "A Compact UWB Antenna Design Using Rounded Inverted L-Shaped Slots and Bevelled Asymmetrical Patch," Progress in Electromag. Research C, vol. 80, pp. 131-140, 2018.

[8] R. M. Elsagheer, "Study on Bandwidth Enhancement Techniques of Microstrip Antenna," Journal of Electrical Systems and Information Technology, vol. 3, pp. 527-531, 2016.

[9] A. A. Deshmukh, A. A. Desai, S. A. Shaikh, K. A. Lele and S. Agrawal, "Formulations for Hexagon-Shaped Ultra-Wide Band Antennas," International Conference on Communication Technology (ICCT 2015), New York, USA, 2015.

[10] S. Morya, S. Saxena, S. Singh and R. Mohan, "Study and Design of Hexagonal Patch Antenna for UWB System," International Research Journal of Engineering and Technology (IRJET), vol. 4, no.4, 2017.

[11] T. Mandal and S. Das, "Microstrip Feed Spanner Shape Monopole Antennas for Ultra-Wide Band Applications," Journal of Microwaves, Optoelec.and Electromag. App., vol. 12, no. 1, pp. 15-22, 2013.

[12] T. Nilakhe and V. V. Patil, "Bandwidth Enhancement through Fractal Nature of Hexagonal Microstrip Patch Antenna," International Journal of Science, Engineering and Technology Research (IJSETR), vol. 5, no. 6, pp. 1936-1940, 2016.

[13] B. Premalatha, M. V. S. Prasad and M. B. R. Murthy, "Compact Hexagonal Monopole Antenna," Indian Journal of Science and Technology, vol. 10, no. 19, 2017.

[14] D. Aissaoui, L. M. Abdelghani, N. Boukli-Hacenand T. A. Denidni, "CPW-Fed UWB Hexagonal-Shaped Antenna with Additional Fractal Elements," Microwave and Optical Technology Letters, vol. 58, no. 10, pp. 2370-2374, 2016.

[15] G. Al-Jaafreh, Hexagonal Printed Circuit Antenna for Wireless Communication, M. Sc. Thesis, Electrical Engineering Department, The University of Jordan, April 2018.

[16] D. M. Pozar, Microwave Engineering, Third Edition, New York, Wiley, 2005.

[17] N. Chattoraj, "Effect of Ground Plane on a Tapered U Slot Ultra-Wideband Antenna," International Journal of Computer Applications (IJCA), Foundation of Computer Science (FCS), no. 2, pp. 8-11, 2013.

[18] C.- C. Lin, K.-Y. Kan and H.-R. Chauang, "A 3-8 GHz Broadband Planar Triangular Sleeves Monopole Antenna for UWB Communication," IEEE Antenna and Propagation Society Inter. Sym., USA, 2007.

[19] M. J. Amman, "Control of the Impedance Bandwidth of Wideband Planar Monopole Antennas Using a Bevelling Technique," Microwave and Optical Technology Letters, vol. 30, no. 4, pp. 229-232, 2001.

[20] S. Jacob and P. Mohanan, Design and Analysis of Printed UWB Antenna with Dual Band-notched Characteristics, Doctoral (PhD) Thesis, Cochin University of Science and Technology, 2015.

ملخص البحث:

في هذه الورقة ، يُقدم هوائي مسطح محسن سداسي الشكل من أجل تطبيقات النطاق الترددي فائق العلوّ. تم تصميم رقعة سداسية بسِتّ قنوات دائرية عند رؤوسها على مادة أساس من نوع FR4، مع خط تغذية مثلث مستدقّ على شكل شريط صغير مقاومته 50 أوم، ومع سطح أرضي مستوٍ مشطوف جزئياً يحتوي على خمسة أكمام نصف دائرية. وقد تم استقصاء التصميم المقترح باستخدام مُحاكٍ بنيوي للترددات العالية؛ إذ أظهرت النتائج المتعلقة بالمحاكاة والقياس فيما يخصّ معامل التشتت S11 (معامل الانعكاس) مواءمة جيدة للمانعة في النطاق الترددي 3-27.57 جيجاهيرتز، محققة فقد رجوع يساوي أو يزيد على 10 ديسيبل مع نطاق ترددي نسبته 160.75%. وقد تحقق كسب عالٍ وفعالية عالية، إضافة الى نمط إشعاع مماثل لنمط إشعاع هوائي ثنائي الأقطاب في المستوى E، مع إشعاع في جميع الاتجاهات في المستوى H.

ENHANCING THE ACCURACY OF SONBOL'S ARABIC ROOT EXTRACTION ALGORITHM

Nisrean Thalji[1], Nik Adilah Hanin[1], Zyad Thalji[2] and Sohair Al-Hakeem[3]

(Received: 11-Sep.-2018, Revised: 5-Oct.-2018, Accepted: 25-Oct.-2018)

ABSTRACT

Root extraction is an important primary process in most Arabic applications, such as information retrieval systems, text mining, text classifiers, question answering systems, data compression, indexes, spelling checkers, text summarization and machine translation. Any weaknesses of root extraction will affect negatively the performance of these applications. Sonbol's Arabic root extraction algorithm achieves high accuracy of performance and gives new classification for Arabic's letters which minimizes the affix ambiguity. The comparison and testing of the existing Arabic root extraction algorithms on unify datasets shows that they still need some enhancements. Arabic root extraction is mainly based on using patterns, where as much as the algorithm has patterns as much as the accuracy is better. In this study, we improve Sonbol's Arabic root extraction algorithm, by enhancing its rules and increasing its patterns. We use 4320 patterns to extract the roots, which is the largest patterns' list extracted by Thalji's corpus. We test the new algorithm on Thalji's corpus that contains 720,000 word-root pairs. This corpus is mainly built to test and compare Arabic root extraction algorithms. The new algorithm is compared with Sonbol's Arabic root extraction algorithm. The algorithm of Sonbol et al. achieves an accuracy of 68%, whereas the new algorithm achieves an accuracy of 92%.

KEYWORDS

Arabic root extraction algorithm, Stemming, Arabic language processing.

1. INTRODUCTION

Arabic language is one of the most used Semitic languages. Semitic languages are spoken in a number of regions that were common in distant times in many regions of Africa and Asia over many decades. Some of these languages are not used now, such as Akkadian, Assyrian and Babylonian and some languages are still used nowadays, such as Arabic, Hebrew and Syriac. Semitic languages are a branch of the Afro-Asiatic language family originating in the Middle East (Bennett, 1998) [1].

In Arabic, vowels are used to ensure the exact meanings of words. If the word is non-vocalized, in many cases it is an ambiguous word and then we need to read the sentence and sometimes the whole text to understand the exact meaning. These vowels are written above or under the letter. Table 1 shows vowels in Arabic and corresponding letter/s in English. Some words in non-vocalized texts may have more than one meaning (ambiguous words). So, they have different roots. For example, the non-vocalized Arabic word "والدين/WALDN" has three possible words "وَالِدَيْن /WALEDAYN", "وَالدَّيْن/ WA ADDAYN" and "وَالدِّين/ WA ADDEEN". And then, the possible roots are "وَلَد/WALAD" (son), "دَين/DAYN" (debt) and "دِين/DEEN" (religion). Another example is the non-vocalized Arabic word "كتب/KTB" which has three possible interpretations: " كَتَبَ/ KATABA" (he wrote), "كُتِبَ/ KUTIBA" (has been written), and "كُتُبٌ/ KUTUBUN" (books) [2]. We converted the Arabic letters and words into Latin characters (uppercase), so that the reader who is not familiar with Arabic texts can read it with ease. This way, the reader will be able to read words as the way they sound phonetically.

Root extraction is an important primary process in most Arabic applications, such as information retrieval systems, text mining, text classifiers, question answering systems, data compression, indexes, spelling checkers, text summarization and machine translation.

1. N. Thalji and N. Hanin are with Department of Computer Engineering, School of Computer and Communication Engineering, University Malaysia Perlis, Malaysia. Emails: nnthalji1980@gmail.com, adilahhanin@unimap.edu.my
2. Z. Thalji is with Department of Management Information System, Imam Abdulrahman Bin Faisal University, Kingdom of Saudi Arabia. Email: zythalji@iau.edu.sa
3. S. Al-Hakeem is with Department of Computer Science, Ajloun National University, Jordan. Email: drsohair@gmail.com

Table 1. Vowels in Arabic and corresponding letter/s in English.

No.	Vowels in Arabic	Corresponding letter/s in English
1	◌ِ	E
2	◌ُ	O
3	◌َ	A
4	◌ْ	No letter
5	◌ٍ	En
6	◌ٌ	Un
7	◌ً	An
8	◌ّ	Duplicate the letters

Therefore, many Arabic root extraction algorithms are presented in many different studies that tried to algorithm of Sonbol, Ghneim and Desouki [3]. The algorithm of Sonbol et al. comes after the algorithm of Khoja and Garside [4], which is well-known in extracting Arabic roots. Khoja and Garside algorithm’s accuracy amounted to 95% when they tested their algorithm with their own dataset. Sonbol et al. tried to improve Arabic root extraction algorithms in order to increase the percentage of accuracy. The algorithm of Sonbol et al. accuracy amounted to 98% when they tested their algorithm using their own dataset.

Al-Shawakfa, Al-Badarneh, Shatnawi, Al-Rabab’ah and Bani-Ismail [5] made a comparison study of existing Arabic root extraction algorithms. This comparison included the algorithm of Sonbol et al., Khoja and Garside algorithm and other algorithms. This comparison was conducted in a unified dataset, in order to evaluate these algorithms fairly. Khoja and Garside algorithm’s accuracy was 34%, whereas Sonbol algorithm’s accuracy was 24%. Variance in the accuracy values is due to the differences of datasets that were used in the testing process. The study by Al-Shawakfa et al. revealed that existing Arabic root extraction algorithms still need more improvement. It also presented some weaknesses of Khoja and Garside’s algorithm and the algorithm of Sonbol et al. In this study, we continue completing Sonbol’s work by improving their algorithm.

There are three different approaches to extract the roots of the word; rule-based approach, lookup table approach and statistics-based approach. Recently, most of the root extraction algorithms are rule-based approach [5]. This approach mainly has two parts; the lists of affixes (roots and patterns) and the rules. Each Arabic root extraction algorithm tried to enhance the lists and/or the rules.

However, most algorithms suffer from the following problems:

- There are neither standard dataset nor complete lists of Arabic affixes, patterns and roots. Each work collects or generates their own dataset or lists. Most of the lists which contains the affixes, patterns, roots and lists that they used in each work are not publicly available. They just wrote samples of these lists. As a result, every time a new root extraction method is proposed, the researchers need to collect their own data or generate their own list independently. This will cause overlapped works, where each work is trying to solve the same problem instead of improving each other’s work, which resulted a waste of time and resources. In addition, the lists used might have significant difference in terms of number of words, which will make it difficult for researchers to fairly compare the performance of existing works. Therefore, in recent works, the researchers tried to extend these lists by adding new contents [6].
- Arabic has a complex structure, which makes it difficult to extract the roots [7]. All Arabic root extraction algorithms suffer from affixes’ ambiguity, so that it is difficult to distinguish between affix letters and root letters.

This work focus on solving these problems. The structure of this paper is organized as follows; in Section 2, different related previous studies and their drawbacks are discussed. Section 03 describes the proposed methodology which includes the details of each process. Section 0 explains the experimental implementation of our algorithm and its evaluation process. Section 5 concludes the main points of the paper and gives some future directions.

2. Previous Studies

In this section, we give a brief overview of previous rule-based Arabic root extraction algorithms. Khoja and Garside and Garside algorithm [4] is a very popular rule-based Arabic root extraction algorithm.

Khoja and Garside and Garside algorithm reported 96% accuracy of their stemmer using newspaper text. Al-Shalabi [8] presented an Arabic root extraction algorithm, which is a rule-based algorithm that is used to extract trilateral roots of Arabic words. This algorithm has been tested on a corpus of 72 abstracts 10582 words from the Saudi Arabian National Computer Conference, where its accuracy was about 92%.

Al-Kabi and Al-Mustafa algorithm [9] is based on affix removal. They tested their algorithm on small data sets containing 1,827 words. The system failed to analyze 55 words, since their patterns are unknown. This failure was mostly due to foreign (Arabized) words. The system enables to analyze the rest 1,772 words and achieved 91% of accuracy.

Sonbol, Ghneim and Desouki [3] algorithm is a rule-based root-extraction algorithm, the principal idea of which is based on encoding Arabic letters with a new code that preserves morphologically useful information and simplifies its capturing toward retrieving the root. They conducted their experiments using two different corpuses. The first corpus consisted of lists of word-root pairs 167,162 pairs. The second corpus was a collection of 585 Arabic articles from different categories (policy, economy, culture, science and technology and sport). This corpus consisted of 377,793 words. In general, the accuracy was about 96%-98%.

Another work is Ghwanmeh, Al-Shalabi, Kanaan, Khanfar and Rabab’ah algorithm [10], which is a rule-based algorithm used to find trilateral Arabic roots. According to Ghwanmeh et al.., their algorithm has only failed to analyze words that are normally foreign, irregular or do not have trilateral roots. A corpus of 242 abstracts from the Proceedings of Saudi Arabian National Computer conference in machine-readable form was used in the testing procedure. The set of abstracts was chosen randomly from the corpus for analysis. The results obtained showed that the algorithm extracts the correct roots with an accuracy rate up to 95%. Many algorithms have been conducted under this type, like the Kchaou and Kanoun algorithm [11], El-Defrawy, El-Sonbaty and Belal algorithm [12] and Ayedh and Guanzheng algorithm [13].

Also, many morphological analyzers have been conducted to properly provide maximum morphological information on Arabic words, such as the proclitic, the prefix, the lemma, the suffix, the stem, the root, the enclitic, the tag and the pattern. One of them is MADAMIRA [14], a morphological analyzer that provides many information on Arabic words. MADAMIRA combines two morphological analysis systems; MADA [15], [16] and [17] and AMIRA [18].

In addition, Al-Khalil Morphological System 2 [19] is a recent morphological analyzer that provides many information on Arabic words. It deals with vocalized and non-vocalized Arabic words. It overcomes many errors of the previous system Al-Khalil Morphological System 1.

In general, all rule-based Arabic roots extraction algorithms share seven processes, which are: normalization, removing prefixes and suffixes, matching the word against patterns, extracting the roots from the patterns, comparing the roots with the roots' list, returning the extracted roots and finally making enhancement to extract the correct roots, as shown in Figure 1.

The difference between one algorithm and the others is in the details of each process. Also, every algorithm has different lists of prefixes, suffixes, roots and patterns. The main problem is in process two and process four. In process two, in many cases, the algorithms remove the matched prefixes and suffix letters, but these letters are part of the root. So, the result is a wrong root. Our proposed solution to this problem is done by not removing prefixes or suffixes and collecting more rules to reduce affix ambiguity. In process four, the algorithms match the word against patterns to extract the root. The main problem is the limited number of patterns that are collected till now. The extraction accuracy will improve if the algorithm can test as many as existing word patterns. Our proposed solution to this problem is using longer pattern lists. The proposed algorithm uses Thalji’s pattern [6], which is the most recent list. This list is automatically generated from most of the Arabic dictionaries and contains (4320) patterns, which is the longest list discovered until now. These patterns contribute to enhancing the accuracy of Arabic root extraction algorithms.

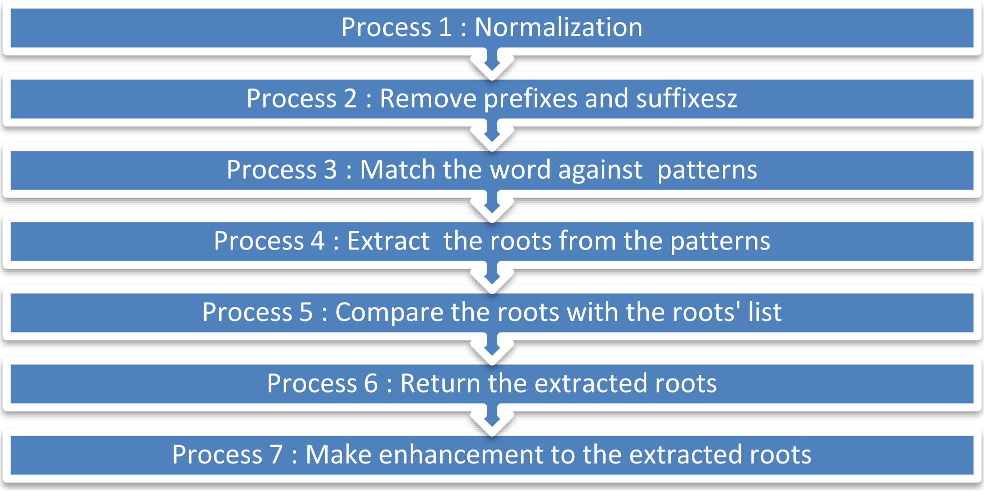

Process 1 : Normalization
Process 2 : Remove prefixes and suffixesz
Process 3 : Match the word against patterns
Process 4 : Extract the roots from the patterns
Process 5 : Compare the roots with the roots' list
Process 6 : Return the extracted roots
Process 7 : Make enhancement to the extracted roots

Figure 1. Main processes in rule-based Arabic root extraction algorithms.

3. METHOD

The root is the base form of the word that gives the main meaning of the word. In this section, the methodology for the proposed Arabic root extraction algorithm is explained. The proposed algorithm is an enhancement of the algorithm of Sonbol et al. by increasing the rules and the lists to find all possible roots of the word.

3.1 Normalization

The normalization steps for the algorithm of Sonbol et al. are as follows:

- Removing the kasheeda symbol ("_").
- Removing the diacritics.
- Replacing the Hamza’s forms (ء, آ, ؤ, إ, ئ) with the letter (أ).

In this section, we extend Sonbol’s normalization process by the following steps:

- Removing the punctuations.
- Removing the non-letters.
- Duplicating any letter that has the Shaddah: “ّ" symbol.

3.2 Encoding

In this step, Sonbol coded the Arabic letters based on six symbols {O, P, S, PS, U, A}, representing six groups of letters each of which shares certain characteristics:

O: Original letters. These letters are surely part of the root. They are: (“ث”, “THAA”), (“ج”, “JEEM”), (“ح”, “HAA”), (“خ”, “KHAA”), (“د”, “DAL”), (“ذ”, “THAL”), (“ر”, “RAA”), (“ز”, “ZAY”), (“ش”, “SHEEN”), (“ص”, “SAD”), (“ض”, “THAD”), (“ظ”, “DAA”), (“ط”, “TAA”), (“ع”, “AYEN”), (“ق”, “GAF”), (“غ”, “GAYEN”). This means that if the word contains one or more of these letters, these letters should be part of root’s letters.

P: Prefix letters ”ل, س, ف, ب“ (BAA, FAA, SEEN, LAM). These letters should be treated as part of the root word if they appear in a different part of the word other than the prefix part. Otherwise, these letters are considered ambiguous letters (can be part of the root word or added letters to the root word). If these letters are ambiguous letters, the algorithm initially considers them as prefix letters. There is a possibility for these letters to become (O) letters (root’s letters) in the next steps.

S: Suffix letter (“ه’’, Haa). This letter should be treated as part of the root word if it appears in a different part of the word other than the suffix part. Otherwise, this letter is considered as an ambiguous letter. If this letter is an ambiguous letter, the algorithm initially considers it as suffix letter. There is a possibility for this letter to become (O) letter (root’s letter) in the next steps.

PS: Prefix-Suffix letters “ك، م، ن”(KAF, MEEM, NOON). These letters can appear only on both sides of the word; i.e., in the suffix part or in the prefix part. These letters should be treated as part of the root word if they appear in a different part of the word other than the prefix-suffix part. Otherwise, these letters are considered ambiguous (can be part of the root word or added letters to the root word). If these letters are ambiguous letters, the algorithm initially considers them as prefix-suffix letters. There is a possibility for these letters to become (O) letters (root’s letters), (P) prefix letters or (S) suffix letters in the next steps.

U: Uncertain letters (”ا, و, ي, ت ”(TAA, YAA, WAW, ALEF)). These letters can appear anywhere in the word. It is not possible to verify whether these letters are part of the root word letters. Several cases are associated with these letters as they may change, omit or convert from one letter to another during the derivation process following well known Arabic rules “EBDAL and EALAL”. For example, the Arabic word” قيل/KEEL” (It was said) and its root “قول”, the letter “و” is converted into “ي” during the derivation process.

A: Added letter (“ة “only (TAA MARBUTA)). This letter is always considered an additional letter. This letter is always deleted.

3.3 Some Improvements to the Last Coding

In this section, Sonbol et al. added some improvements to the last coding, by applying the following conditions:

- The letter “ب” (BAA) is a prefix letter if it is situated in the first three letters; otherwise, it is an original letter (part of the root word letter). This rule means that if the letter “ب” (BAA) is situated among the first three letters, it is an ambiguous letter. On the other hand, it will be part of the root word if it appears in place other than the first three letters of the word. For example, with the words “ بأمرك, وبالصـــعيد”(BEAMRK, WBLSA”EED), the letter “ب” (BAA) is situated among the first two letters. So, it’s an ambiguous letter. Initially, the algorithm considers it as a prefix letter. Then in the next steps, it may change to (O) letter (root’s letter). Another example is the word “افبالباطل/AFBLBATL”; this word has the letter “ب” (BAA) appearing twice in the word. The first “ب” (BAA) letter is an ambiguous letter, as it appears as the third letter in the word. Initially, the algorithm considers it as a prefix letter. Then, in the next steps, it may change to (O) letter (root’s letter). The second “ب” (BAA) will be considered part of the root word, as it appears as the sixth letter of the word.

- The letter “ف” (FAA) is a prefix letter if it appears among the first two letters; otherwise, it is an original letter. This means that if the letter “ف” (FAA) appears among the first two letters of the word, it is an ambiguous letter. If the letter “ف” (FAA) appears in a place other than the first two letters, it is part of the root word. Initially, the algorithm considers it as prefix letter. For example, in the words “فهد, فاسـتأجرتها, افغير” (FHD/FASTAJRTHA, AFGYR), the letter “ف” (FAA) appears among the first two letters of the word. So, it’s an ambiguous letter. Initially, the algorithm considers it as a prefix letter. It may change to (O) letter (root’s letter) in the next steps of the algorithm following certain rules. Another example; the words “تسافهت / TSAFHT” ,“متفائلين/ MTFAELEEN” and “وأتفكت/ WETFKT”; the letter “ف” (FAA) is considered to be part of the root word as it appears in places other than the first two letters of the word.

- The letter “س” (SEEN) is a prefix letter if it is followed by one of the letters " أ, ن, ي, ت" (HAMZA, NOON, YAA AND TAA); otherwise, it is part of the root word. For example, the words “سـتبقى/ STBKA, سـاتغير/ SATGYR ,سـنتواجد /SNTWAJD, سـينجلي /SYNJLE”; the letter “س” (SEEN) is an ambiguous letter, as it is preceding one of the letters " أ, ن, ي, ت" (HAMZA, NOON, YAA AND TAA). Initially, the algorithm considers it as a prefix letter. It may change to (O) letter (root’s letter) in the next steps of the algorithm. Another example is with the words “كالأسـد, مأسـور” (KALASD, MASOR); the letter “س” (SEEN) is considered part of the root word, as it is not preceding one of the letters " أ, ن, ي, ت" (HAMZA, NOON, YAA AND TAA).

- The letter “ل” (LAM) is considered a prefix letter if it appears among the first five letters of the word; otherwise, it is part of the root word.

- The letter “ه” (HAA) is considered a suffix letter if it appears among the last three letters of the word; otherwise, it is part of the root word.

- The letter “ك” (Kaf) is considered a prefix letter if it appears among the first three letters of the word; otherwise, it is a suffix letter.

If the algorithm of Sonbol et al. finds three O-Letters (or more) in the encoded word, these letters are considered root letters and the algorithm is terminated. However, in this work, the enhancement of the algorithm of Sonbol et al. is to continue searching for other possible roots and for longer roots (more than three letter root word).

3.4 Applying Transformation Rules

In this section, the algorithm of Sonbol et al. applies transformation rules between groups to obtain a maximum number of original letters. Transformation rules are mentioned below:

- R1) Change each (P) after (O) to (O).

For example, with the word “ضيوف/ DYUF”, “ض” (DAA) is an (O) letter, “ف” (FAA) is a (P) letter; in this case, (P) comes after (O). So, “ف” (FAA) is changed to (O) letter, which means that it should be one of the root’s letters. Until now “ف ,ض” (FAA, DAA) are part of the root’s letters.

- R2) Change each (S) before (O) to (O).

For example, with the word “الهداية/ ALHDAYH”, “د/ DAA” is an (O) letter, “ــهــ” is a (S) letter; in this case, (S) comes before (O). So, “ـهـ/HAA” is changed to (O) letter, which means that it should be one of the root’s letters. Until now “د ,ـهـ/ DAA, HAA” are part of the root’s letters.

- R3) Change each (PS) before (P) to (P).

For example, with the word “كالسيوف/KALSOYUF”, “كـ/KAA” is a (PS) letter, “ســ/SEEN” is a (P) letter; in this case, (PS) comes before (P). So, “كـ/KAA” is changed to (P) letter, which means that it should be one of the root’s letters or prefix letters, but not a suffix letter.

- R4) Change each (PS) before (O) to (P).

For example, with the word “منتقمون/ MNTKMON”, “ـنـ/ NOON” is a (PS) letter, “ـقـ/ KAA” is an (O) letter; in this case, (PS) comes before (O). So, “ـنـ/NOON” is changed to (P) letter, which means that it should be one of the root’s letters or prefix letters, but surely not a suffix letter.

- R5) Change each (PS) after (S) to (S).

For example, with the word “منتهك/ MNTHK”, “ك/KAF” is a (PS) letter, “ـهـ/HAA” is a (S) letter; in this case, (PS) comes after (S). So, “ــك/KAF” is changed to (S) letter, which means that it should be one of the root’s letters or suffix letters, but surely not a prefix letter.

- R6) Change each (PS) after (O) to (S).

For example, with the word “علمك/ELMK”, “ك/ KAAF” is a (PS) letter, “ــع/A’A” is an (O) letter; in this case, (PS) comes after (O). So, “ك/ KAAF” is changed to (S) letter, which means that it should be one of the root’s letters or suffix letters, but surely not a prefix letter.

- R7) Change each (P) after (S) to (O).

For example, with the word “التهبت/ ELTHBT”, “ب/ BAA” is a (P) letter, “ــهــ/ HAA” is an (S) letter; in this case, (P) comes after (S). So, “ب/ BAA” is changed to (O) letter, which means that it should be one of the root’s letters. Until now, “ب ,ـهـ” (BAA, HAA) are root’s letters.

- R8) Change each (S) before (P) to (O).

For example, with the word “البهتان/ ALBHTAN”, “ـهـ/HAA” is an (S) letter, “ـبـ/BAA” is a (P) letter; in this case, (S) comes before (P). So, “ـــهـــ/HAA” is changed to (O) letter, which means that it should be one of root’s letters. Until now, “ـهـ ,ب” (BAA, HAA) are the root’s letters.

As in the previous step, if the algorithm of Sonbol et al. has three O-letters in the encoded word, these letters are considered root letters and the process will terminate here. However, the enhancement of the algorithm of Sonbol et al. is to continue searching for other possible roots and for longer roots, with more than three root length.

3.5 Extracting All Possible Patterns of the Word

The algorithm of Sonbol et al. uses the idea of traditional algorithms, but with the aid of the encoded word. Traditional algorithms store lists of Arabic prefixes, suffixes and patterns. These algorithms delete prefixes and suffixes, then use the pattern to extract the root from the reminder. The enhancement of this process is done by using larger lists' content and not removing the prefixes or suffixes, but applying the patterns. It's worth to mention that the presented algorithm assumes that patterns are composed of three elements: prefix, stem and suffix. One of the problems that were experienced by most of the previous algorithms is that they delete clitics before comparing with patterns. And in many cases, these clitics are parts of roots and not clitics. For example, with the words "كـالحون, التقى"(Altka, kalehon), removing "كال, ال" (AL, KAL) will give these roots "حون, تقى"(TKA, HON), ignoring other possible roots "كلح, لقى"(LKA, KLH). In this section, we use the pattern's list of Thalji's corpus that was automatically extracted [6]. Up until now, this corpus contains the largest list of 4,320 patterns, which is the most appropriate list to be used in this work. Previous algorithms have used short lists that were manually collected. In addition, they did not publicly publish all the lists' contents. Thalji's patterns are listed in appendix A, so that future researchers can benefit from them.

In this step, we compare the word with the Thalji's list of patterns and return all matched patterns. For example, for a word like "فهد/ FHD", the algorithm found two original letters, "ـــهـــ, د"(DAA, HAA). Next, the word is compared to the list of patterns and all matched patterns were returned. The word "فهد/ FHD" matches the pattern"فعل / FA'L". The word "فهد/ FHD" is the first possible root. Another example is the word "البحر/ ALBAHER", where the algorithm just finds two original letters, which are"حـــ, ر " (RAA, HAA). The word is compared to the list of patterns and all matched patterns were returned. The word "البحر/ ALBHR" matches the pattern "الفعل/ ALFA'L". The word "بحر/ BHR" is the first possible root. Also, this word matches the patterns "افعل, فعلل"(FA"LL/ EFA"L), then "ابحر, البحر" (ALBHR/ ABHR) are also possible roots.

3.6 Extracting All Possible Roots for the Word

All possible roots are found by matching the words against the list patterns. All the possible roots that match the patterns are extracted after finding all possible patterns.

3.7 Solving the Problems with Ealal Rules and Ebdal Rules

When we have a weak letter (ALEF, YAA AND WAW), we replace this letter with the two other letters and check if the result is a valid root. If so, we add this root to the possible roots. For example, in the word "قـال/ KAL", the algorithm replaces "ا/ ALEF" with "ي/ YAA" and "و/ WAW". So, "قول, قيـل" (KEEL, KAWL) are possible roots.

3.8 Minimizing Possible Roots by Comparing them with Roots' List

In this section, we use the roots' list of Thalji's corpus that was automatically extracted from most well-known Arabic dictionaries. It is the largest roots' list found till now with 12,000 roots. This list is longer than the list that is used in Ababneh, Al-Shalabi, Kanaan and Al-Nobani stemmer [20]. It has about 11,347 roots. The distribution of roots for these two different lists is shown in Table 2.

Table 2. The distribution of the roots for two different lists.

Roots	List of Thalji's corpus	List of Ababneh, Al-Shalabi, Kanaan and Al-Nobani stemmer
Two-letter roots	500	115
Three-letter roots	7912	7198
Four-letter roots	3180	3739
Five-letter roots	360	295
Six-letter roots	48	0

The presented algorithm uses Thalji’s list to minimize the possible roots, whereas the algorithm of Sonbol et al. used a short list of roots. For example, in the word “البحر/ ALBHR”, the possible roots are ” ,بحر البحر ,ابحر” (BHR, ALBHR, ABHR), while the roots “البحر ,ابحر” (ALBHR, ABHR) are excluded, because they are not found in the roots’ list.

4. Experiment and Evaluation

In this section, the presented algorithm is compared with other algorithms with the same approach, which is the rule-based approach. These algorithms are Khoja and Garside’s Arabic root extraction algorithm and Sonbol’s Arabic root extraction algorithm. In addition, the presented algorithm is also compared with one of the most recent morphological analyzer systems, which is Al-Khalil Morphological System 2. Al-Khalil Morphological System 2 gives maximum morphological information of Arabic words, such as the proclitic, the prefix, the lemma, the suffix, the stem, the root, the enclitic, the tag and the pattern. A complete comparison was conducted between the algorithms on Thalji’s corpus in terms of accuracy. Thalji’s corpus is an automatic corpus that is built from ten old Arabic dictionaries; this corpus is mainly built to test and fairly compare Arabic root extraction algorithms. This corpus contains 720,000 word-root pairs, which helps to avoid the interference of a human expert normally needed to verify the correct roots of each word used in the testing or comparison process. Moreover, this corpus has more than 4,320 types of words derived from 12,000 roots. Therefore, the list used in this experiment is more comprehensive compared to previous works.

The result of testing shows that the accuracy of Khoja and Garside’s algorithm was 63%, the accuracy of the algorithm of Sonbol et al. was 68%, the accuracy of Al-Khalil Morphological System 2 was 75%, whereas the accuracy of the presented algorithm was 92%. Figure 2 shows the performance accuracy of all compared algorithms.

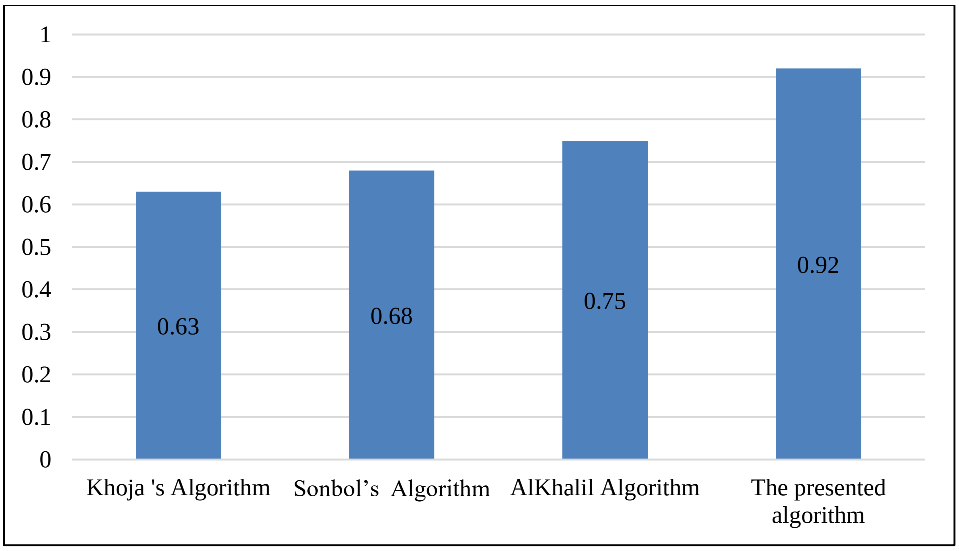


Figure 2. Accuracy of the algorithm of Khoja and Garside, the algorithm of Sonbol et al., the algorithm of Al-Khalil and the presented algorithm.

The main problems of Khoja and Garside’s algorithm are that it does not consider many roots, prefixes, suffixes and patterns. It suffered from affix ambiguity problems. In addition, it returned just one solution for non-vocalized words, ignoring other possible solutions. Besides, it replaced a vowel letter with the letter “و” that sometimes returns a root that is not related to the derivation word. Finally, it produced wrong roots being unsuccessful to extract roots for derivation words that contain the “ابدال/ EBDAL” rule.

The main problems of the algorithm of Sonbol et al. arise if the root does not contain any constant letter, if the root does not start with a constant letter or if the root contains only one constant letter. Also, it does not consider many roots, prefixes, suffixes and patterns. In addition, it returned just one solution for non-vocalized words, ignoring other possible solutions.

The main problems of Al-Khalil Morphological System 2 are that it failed to analyze some words, which were about 25% of the input words. Table 3 shows a sample of these words. For example, the words

“شـــجير, جــارودة, مــدارح, مقــاصـــر, ارصـــدت” (ARSDT, MKASR, MDARJH, JARODH, SHJER) are straightforward to find the roots, because they contain three original letters, but the algorithm fails to analyze them. And in some cases, it returns non-acceptable roots. For example, with the word “التقى/ALTKA”, the generated roots are “وقى, لقي”(WKA,LKE), where the root “وقى/WKE” is not an acceptable root, because “ل/LAM” letter cannot be an infix letter. Also, the algorithm matches the word to the wrong pattern, which is “فعاة/ FA”AH”. In addition, the algorithm fails to find all possible roots of non-vocalized words, like the word “التقى/ ALTKA”, where it doesn’t return the possible root “تقى/TKA”.

The main problems of the proposed algorithm are that it fails to extract the root of derivation words with one letter length. In Arabic language, there are some few derivation words with one letter length, like "قِ, رِ, ع"(KE, RE, A’E). These derivation words are derived from a weak root with a length of three letters and these weak letters are deleted during the derivation process.

Table 3. A sample of unanalyzed words by Al-Khalil Morphological System 2.

عفير	عواطف	قضيم	مدارج	مضمار	مقاصر	ارصدت	دواعب	البضيع	سخريا
البنود	البهيم	التساخين	التقدمية	التماجيد	التواجد	الشجير	الجارودة	الجبابرة	الجريم
الجفلى	الجورب	الحريصه	الحكومات	الحوارد	الخبيص	الخليف	الدبسه	الرجوليه	الرجيع
الرحيق	الرعاديد	الزغاليل	الزهرة	السبله	السميد	الشريم	الصناديد	الضراغمه	الطوارف
العريكة	العنزي	اللطيم	المراشد	المغبرة	بالمناقيش	جحوشا	دهريا	رصيد	زهيره

Another case in which the proposed algorithm still fails is to find the root of derivation words as in the word “درهم/ DERHAM”. The algorithm produces these roots “ودر, دور, دري, درأ, درر "(WDR, DWR, DRE, DRA, DRR). In this derivation word’s matter, the algorithm finds two constant letters in the derivation word and tries to find the third constant letter in order to produce trilateral roots. However, the proposed algorithm is stopped to continue looking for the fourth one.

The proposed algorithm and Al-Khalil Morphological System 2 produce more than one possible root of the derivation words. In contrast, Khoja and Garside’s algorithm and the algorithm of Sonbol et al. produce just one root. In this section, the proposed algorithm and Al-Khalil Morphological System 2 are evaluated in terms of the average of possible roots per word and the number of processed words per second. The result is summarized in Table 4.

Table 4. Comparison between the proposed algorithm and Al-Khalil Morphological System 2.

The algorithm	The average of possible roots per word	The number of processed words per second
The proposed algorithm	3	101
Al-Khalil Morphological System 2	5	105

5. Future Work

The presented algorithm particularly contributes to enhancing the algorithm of Sonbol et al. by increasing its rules and extending its lists’ contents by using Thalji’s lists. The presented Arabic root extraction algorithm is compared with Khoja and Garside’s Arabic root extraction algorithm, Sonbol’s Arabic root extraction algorithm and Al-Khalil Morphological System 2. The testing and comparing processes are conducted on Thalji’s corpus, where the result of testing shows that the accuracy of Khoja and Garside’s algorithm was 63%, whereas the accuracy of the algorithm of Sonbol et al. was 68%, the accuracy of Al-Khalil Morphological System 2 was 75%. The presented algorithm achieved an accuracy of 92%.

In future, we plan to enhance the accuracy of the presented algorithm, overcome some weakness points and enhance the result to return just the exact root word. In order to implement this, the system must have the ability to understand the whole sentence or sometimes the whole paragraph.

REFERENCES

[1] W. Abo Thuaaib, History of Sematic Languages, Lebanon: Darul Kalam for Pub. and Printing, 2016.

[2] A. Al-Taani and S. A. Al-Rub, "A Rule-based Approach for Tagging Non-vocalized Arabic Words," The International Arab Journal of Information Technology, vol. 6, no. 3, pp. 320-328, 2009.

[3] R. Sonbol, N. Ghneim and M. S. Desouki, "Arabic Morphological Analysis : A New Approach," Information and Communication Technologies: From Theory to Applications, Proc. of the IEEE 3rd International Conference, pp. 1-6, 2008.

[4] S. Khoja and R. Garside, "Stemming Arabic Text," Computing Department, Lancaster Univ., UK, 1999.

[5] E. Al-Shawakfa, A. Al-Badarneh, S. Shatnawi, K. Al-Rabab’ah and B. Bani-Ismail, "A Comparison Study of Some Arabic Root Findings," Journal of the American Society for Information Science and Technology, vol. 61, no. 5, pp. 1015-1024, 2010.

[6] N. Thalji, N. A. Hanin, Y. Yacob and S. Al-Hakeem, "Corpus for Test, Compare and Enhance Arabic Root Extraction Algorithms," International Journal of Advanced Computer Science and Applications, vol. 8, no. 5, pp. 229-236, 2017.

[7] M. Sawalha and E. Atwell, "Comparative Evaluation of Arabic Language Morphological Analyzers and Stemmers," Proc. of COLING 22nd Inter. Conference on Comptational Linguistics, pp. 107-110, 2008.

[8] R. Alshalabi, "Pattern-based Stemmer for Finding Arabic Roots," Information Technology Journal, pp. 38-43, 2005.

[9] M. N. Al-Kabi and R. Al-Mustafa, "Arabic Root-based Stemmer," Proceedings of the International Arab Conference on Information Technology, 2006.

[10] S. Ghwanmeh, S. Rabab'Ah, R. Al-Shalabi and G. Kanaan, "Enhanced Algorithm for Extracting the Root of Arabic Words," Proc. of the 6th International Conference on Computer Graphics, Imaging and Visualization, pp. 388-391, 2009.

[11] Z. Kchaou and S. Kanoun, "Arabic Stemming with Two Dictionaries," IEEE International Conference on Innovations in Information Technology, pp. 688-691, 2008.

[12] M. El-Defrawy, Y. El-Sonbaty and N. Belal, "A Rule-based Subject-correlated Arabic Stemmer," Arabian Journal for Science and Engineering, vol. 41, no. 8, pp. 2883-2891, 2016.

[13] A. Ayedh and T. Guanzheng, "Building and Benchmarking Novel Arabic Stemmer for Document Classification," Journal of Computational and Theoretical Nanoscience, vol. 13, no. 3, pp. 1527-1535, 2016.

[14] A. Pasha, M. Al-Badrashinyy, M. Diaby, A. El Kholy, R. Eskander, N. Habash, M. Pooleery, O. Rambow and R. Roth, "MADAMIRA: A Fast, Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic," Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC'14), Japan, 2014.

[15] N. Habash and O. Rambow, "Arabic Tokenization, Part-of-Speech Tagging and Morphological Disambiguation in One Fell Swoop," Proceedings of the 43rd Annual Meeting of Association for Computational Linguistics, pp. 573-580, Association for Computational Linguistics, Michigan, 2005.

[16] N. Habash, O. Rambow and R. Roth, "MADA+ TOKAN: A Toolkit for Arabic Tokenization, Diacritization, Morphological Disambiguation, POS Tagging, Stemming and Lemmatization," Proc. of the 2nd International Conference on Arabic Language Resources and Tools (MEDAR), Egypt, 2009.

[17] N. Habash, R. Roth, O. Rambow, R. Eskander and N. Tomeh, "Morphological Analysis and Disambiguation for Dialectal Arabic," Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, New Orleans, 2013.

[18] M. Diab, K. Hacioglu and D. Jurafsky, "Automated Methods for Processing Arabic Text: from Tokenization to Base Phrase Chunking," Arabic Computational Morphology: Knowledge-based and Empirical Methods, Kluwer/Springer, 2007.

[19] M. Boudchiche, A. Mazroui, M. Bebah, A. Lakhouaja and A. Boudlal, "Al-Khalil Morphological System 2: A Robust Arabic Morpho-syntactic Analyzer," Journal of King Saud University-Computer and Information Sciences, vol. 29, no. 2, pp. 141-146, 2017.

[20] M. Ababneh, R. Al-Shalabi, G. Kanaan and A. Al-Nobani, "Building an Effective Rule-based Light Stemmer for Arabic Language to Improve Search Effectiveness," Int. Arab Jour. of IT vol. 9, no. 4, 2012.

[21] K. Taghva, R. Elkhoury and J. Coombs, "Arabic Stemming without a Root Dictionary," Proc. of the IEEE International Conference on Information Technology: Coding and Computing, pp. 152-157, 2005.

[22] M. Sawalha and E. Atwel, "Corpus Linguistics Resources and Tools for Arabic Lexicography," Proceedings of the Workshop on Arabic Corpus Linguistics (UCREL), 2011.

[23] K. Mezher and O. Nazlia, "A Backpropagation Neural Network to Improve Arabic Stemming," Journal of Theoretical and Applied Information Technology , vol. 82, no. 3, pp. 385-394, 2015.

[24] G. Kanaan, R. Al-Shalabi and M. Sawalha, "Full Automatic Arabic Text Tagging System," Proceedings of the International Conference on Information Technology and Natural Sciences , pp. 258-267, 2003.

[25] E. Al-Shammari and J. Lin, "A Novel Arabic Lemmatization Algorithm," Proceedings of the 2nd Workshop on Analytics for Noisy Unstructured Text Data (ACM), pp. 113-118, 2008.

[26] H. M. Al-Serhan, R. Al Shalabi and G. Kannan, "New Approach for Extracting Arabic Roots," Proceedings of the Arab Conference on Information Technology, pp. 42-59, 2003.

[27] M. N. Al-Kabi, S. A. Kazakzeh, B. M. Abu Ata, S. A. Al-Rababah and I. M. Alsmadi, "A Novel Root-based Arabic Stemmer," Journal of King Saud University-Computer and Information Sciences, pp. 94-103, 2015.

[28] A.-K. N. Al-Kabi, "Towards Improving Khoja Rule-based Arabic Stemmer," Proc. of the IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT), pp. 1-6, 2013.

[29] S. Al-Fedaghi and F. S. Al-Anzi, "A New Algorithm to Generate Arabic Root-pattern Forms," Proceedings of the 11th National Computer Conference and Exhibition, 1989.

[30] F. Abu Hawas and K. E. Emmert, "Rule-based Approach for Arabic Root Extraction: New Rules to Directly Extract Roots of Arabic Words," Journal of Computing and Information Technology, vol. 22, no. 1, pp. 57-68, 2014.

[31] K. Abainia, S. Ouamour and H. Sayoud, "A Novel Robust Arabic Light Stemmer," Journal of Experimental and Theoretical Artificial Intelligence, vol. 29, no. 3, pp. 557-573, 2017.

[32] B. Abuata and A. Al-Omari, "A Rule-based Stemmer for Arabic Gulf Dialect," Journal of King Saud University-Computer and Information Sciences, pp. 104-112, 2015.

[33] S. A. Yousif, V. Samawi, I. Elkabani and R. Zantout, "The Effect of Combining Different Semantic Relations on Arabic Text Classification," World of Computer Science and Information Technology Journal, pp. 112-118, 2015.

[34] G. Kanaan, R. Al-shalabi and M. Sawalha, "Improving Arabic Information Retrieval Systems Using Part of Speech Tagging," Information Technology Journal, pp. 32-37, 2005.

APPENDIX

فعل, الفعلة, يفعل, فعلا, مفعول, المفعول, والفعلة, الفوعل, والفوعل, الفوعلا, والفيعلة, فيعلة, فيعلا, فعولا, فعول, فوعل, فوعلا, الفعول, فعلة, وفعول,
وفاعلون, فاعلة, فواعل, وفاعلات, الفعيلة, وفعلت, الفاعل, المفعل, والفعال, الفعال, فعالة, الفعلاة, فعلون, فعلاة, فعيل, فعال, والتفعل, تفعل, مفعلين,
ومفعل, مفعل, الفعل, والفعل, والمفعل, فعلته, فعلت, المفاعل, فعالها, مفعولة, والمفاعل, مفاعل, لفعلة, وفعيلك, يفاعلك, وتفاعله, بفعلها, فعيلها,
والتفاعل, التفعل, والافتعال, يفتعل, فيعلية, وفيعلة, والفيعل, فيعل, وفعالا, وفعل, وتفعل, الفعالية, والفعالة, ويفاعلها, فاعل, وللفاعل, الفيعلة, وفيعلت,
فاستفعلتها, فعوال, فعوالة, ففعل, بفعوالة, والفواعل,, يفعلون, وتفعلت, تتفعل, الفيعل, وفيعل, الفعلان, والفعلان, فعلى, فعالى, والفاعل, وفعلان
وفاعلة, وفعولت, وفعالة, فاعلا, الفواعل, وفاعل, فواعلا, الفعلا, الفيعالة, بفيعالة, الفياعيل, فيعال, وفيعلتها, فياعلة, وفياعل, بالفيعل, فعيلا, وتفاعلت,
بالفاعلين, متفعل, والفعول, متفعلا, والفعيلة, فعيلة, الفيعله, تفعلوا, تفعلت, وفعيلة, والانفعال, بالفعل, بالمفعول, مفعلا, انفعلت, وانفعلت, تفعلا, وافتعل,
والفعيل, فعلنا, وافتعلت, افتعالا, وفعلة, تفعله, يفعله, كالفعيل, الفعيل, والمتفعل, والمفعول, انفعل, فياعل, فعالا, بفعالته, الفعالة, وفعلا, فعلها, فعيلته,
وفعاليل, فعولهم, ينفعل, وانفعل, فتفعل, فانفعلت, لانفعاله, وفعال, مفاعلا, ففعله, يفعلنه, وفعيل, والفاعلة, والفوعلة, وفعلاوات, تفاعل, وتفاعل, المفعلة,
بالمفاعل, ومتفعل, وفعولته, بالفعول, والفعلولة, الفعاليل, والفعاليل, ومفعلا, فانفعل, منفعل, فعله, ومنفعل, الفعلات, فعلان, فاستفعلته, تفعيلا, يفعلهن,
مفعلة, ومفعلة, والتفعيل, التفعيل, مفعوللات, والمنفعل, استفعلته, ليفعل, مفعال, لفاعل, لفعل, فعلي, فيفعل, بالفعيل, وانفعال, مفعله, والمفعلات,
وفعلات, افتعل, والمفعلة, وفعله, واليفعيل, فاعلته, فعلات, بفعل, الفوعلة, افعل, يفوعل, مفعلات, يفتعله, والفعولة, يفتعلها, وفعيلتك, فعيله, وفعيلا,
فعلن, وفواعله, وفواعل, فعلل, وفعللة, وافعل, والفعلل, فعلك, وافتعلك, فينفعل, يفعلوا, بالفعال, والفاعلان, كفعيل, وفاعله, الافتعال, كفعل, كالفعل,
فعالات, والمفاعلة, يفاعلها, وفعيلته, وتفعلهم, ويفعل, فعلاتهم, لمفعل, لمفعول, فعوله, والفعلوة, فعلوتين, فعلوتان, الفعالي, واتفعله, ومفتعل, مفتعل,
والفعلات, وافعلتها, الفعلى, وتفاعلوا, وفاعلته, ففعلته, وافعلت, يفتعلوا, وفاعلت, يفاعله, فعلوا, والمفعال, وفعلته, ومفعلان, ومفعلانة, فعلتها, تفتعل,
مفعولا, ومفعول, فعالين, الفاعول, فاعول, فعالل, ففاعل, فعولها, ومفعال, تفعلني, مفعاله, وتفعيل, والفعيلي, والفعلى, بالفعلة, بالتفعيل, ومفاعيل,
يتفعل, فعالته, والمفتعل, بالمفتعل, والافعيلال, وافتعلتها, تنفعل, والفعلي, الفعلي, بالفعلي, فالفعول, وفعلي, مفاعيل, مفتتعل, افتعلت, المفاعلة, افتعلوا,
تفاعلوا, افتعلا, وتفعلتها, والفيعول, الفيعول, فالفاعل, افعال, وفعولا, المستفعل, واستفعلت, والفعيله, ومفعولة, يفعلن, متفعلة, فتفعلت, فعلتين, فعاليل,
بمفعال, وفعليه, فعيلان, ويفتعلان, ويتفاعلان, ففعلتم, فعلانا, وفعلى, استفعلت, واستفعل, يتفاعلان, بفاعلة, ففاعله, مفاعلة, يفعلان, واليفعول, يفاعيل,
ويفعول, بفعله, بيفاعيل, اليفاعيل, بالتفاعيل, تفعيل, فعالية, كالفعلة, وفعلتهم, مفعولات, ففعيل, التفعيلا, مفاعلها, المتفعلون, تفعلهم, مفعلان, بفعالة,
وفعلهم, فعالي, مفتعلون, المفتعل, وتفعلكم, فعالاه, بفعالي, وفوعل, فافتعلوا, الفعلانيون, ويفعلون, والفاعلين, فعولة, فعلناه, الفعولا, فاعلتها, وفعلني,

وافتعلته, والفاعلتان, ومنفعله, بتفعيل, المفاعيل, يتفاعل, مفتعلا, وفعلها, فعلاه, وفعلنا, فاعلات, فعيلات, بمفعلين, وفعلك, بفعلول, والفعلول, بالفعلات,
فالفعل, افتعلي, يفتعلون, والفعالات, وافتعال, ومفاعلته, والافعالة, واستفعلته, استفعل, فاعلون, والاستفعال, والمفعولة, تفعلان, فعلاوات, افتعله, افتعال,
والفعولي, الفعولة, فعلانة, الفعاول, فعاول, فعلني, فعاللة, ومستفعل, كفعال, لفعال, فعاله, لفعالة, كتفعل, فعلتك, بالفعلا, فعليات, بفعول, مستفعل,
وفعلول, فعلولة, يفعلك, الفعيلات, الفعالا, فعليه, والفعالي, لمفاعلة, مفاعلوك, تفعال, للفعال, الفعلون, فعلين, وفعلون, فاعلهم, فعلكم, لتفعل, وفعولة,
وفعلانة, فانفعلا, فعلتان, فعلهم, يتفعلن, وافعوللوا, وافعوللت, يستفعل, فافتعل, المفتعلة, والفعلن, فعلنة, بفعليل, الفعليلة, فعليل, والفعله, لمفعلة, فعلناها,
الفاعلتين, فواعلها, فاعلي, الفاعلين, الفاعلة, فعلية, فعلانه, والفعلنة, الفعلنى, بمفاعيل, فعلتهم, فاعلت, ففاعلتها, كفعلك, يفاعل, وفعالتا, نفعل, فيعول,
فيفتعل, الفعلين, انفعالا, فتفعلوا, انفعال, وتفعيلي, فعاويل, الفعاويل, والفعوال, بالمفعلة, افتعاله, وفعلتها, لفعلانات, فعلانات, فعليها, وفعلانها, المفعلا,
فالمفاعل, ومفاعل, المفعالين, والمفعالان, فاعلوا, فيفعله, افعلل, المفعلل, وفعلولة, الفعلول, يفعللها, افتعلته, يفعلنا, المفعلات, افعلت, وفعولها, افتعالي,
للفاعل, المتفعل, وفوعلة, كالمفعيل, الفعلولة, فافعلوها, افعوللا, يفعولل, وافعولل, الفعلية, فعلوني, مفعولل, فعيلتي, فعالتي, وفعيله, الفعلوة, لفعولا,
للفعول, تفعيلهما, فيعلان, فيعلاني, فوعلانية, للفعلى, مفعوله, والفعلانة, والفعالى, كالفعلان, بفاعل, فعيلون, وفعاول, فعولات, والافتعالة, فافتعله,
وفعوله, بالفواعل, ومفعلات, بمفعولل, وافعيلال, فعيلية, كالفعلاة, والفعلاة, كفعلة, يفعولة, لفعلان, والفعلاوة, والفاعول, وفعلوا, وفعيلي, وافتعلوا,
فعولته, والمفعالة, للفعل, كالمفعل, وفعلوه, وفاعلتك, وافتعله, المفعال, فعاليها, بمفعل, ومفتعلا, افعيلالا, وافعال, بالمفعل, اليفعول, لنفعلا, افعلا, يفعلها,
مفعلون, الفعالين, والفعالان, افعوال, المفعيل, مفعيل, فعيولون, فعاييل, الفعيول, فتعلل, فعيلهم, تفعليني, فعلتني, بفعلهم, فوعلة, والفاعلية, مفعالة,
فعليلا, والفعليلة, والفعليل, وافتعالا, افعالا, والمفعلتان, افعله, وفعاله, فيفعلها, ويفتعلها, افعلوا, التفاعل, يفعلهما, الافعال, ليفاعلوا, ليفعلوا, تفعلونني,
ففعلناها, بفعاله, فاعلها, بفعال, وفعالى, وافعولا, وافعول, افعوالا, افعول, الفعولية, فعولية, فعولاة, والفعولية, لافعل, الفعليل, كفعليل, تتفعلل, يتفعلل,
فتفعله, فعولاه, فعلانية, وفعلانيون, وفعلاني, ويفتعل, وتفعلته, ففعيلنا, افتعلهما, فالفعيل, فعلول, تفعلته, بفعلة, تفعلك, بالفعالة, تفعيلة, بالفعولية, وفعلين,
وتفعله, فالفعال, بمفاعل, ليتفاعل, ففعلها, تفعلي, لمتفعل, وفعيلات, المفعولة, وفعلانا, مفاعلي, بالفعيلي, تفعلل, والتفعلل, لفعلي, والفعالتان, يفعلني,
مفعالا, ويفعلها, والفعلا, ومفعله, المفعلون, فعيلك, فعيلي, يتفاعلني, الفعلوانات, والفعلوانة, فعلوان, الفعلوان, وفاعلها, يتفعلون, لتستفعل, والفعلوت,
والفعاليت, وفعليت, وفعالية, وفعلية, ففعلا, مفتعلات, بفعلات, بالفعلى, والفعلين, فياعيل, فيعالة, والفيعال, كفعلان, فعلوة, والفعلون, كالفعال, فيعلت,
تفعالة, والتفعالة, فعلله, والفاعلي, والفعلانية, فعليا, وتفعلوا, فاستفعلوا, المستفعلة, فعليته, فعلاني, وتفعلنا, ففعلت, وفاعولة, فعلناة, فعلنى, الفعليون,
فعلاوان, ويستفعل, تفعيلها, فعاليات, وتستفعل, افتعالل, وافتعالل, اليفعلا, واليفعل, لتفعلها, ليفعلوها, يفعلونها, لفعلك, فعلو, فعاليته, الفعاليات, بمفعول,
وافيعلل, افتعليل, واليفعلات, باليفاعل, يفعلات, واليفعلة, فيعله, والمفعلا, والفاعولة, مفاعلات, ومفاعلا, تفعالا, افتعلل, وفعلاليه, فعلاويه, بالفعالي,
فعللة, الفعالل, بفعلل, فعللا, الفعلل, فتعللة, فعالتهم, فعالتان, الفعالى, لفعلا, يفاعلات, يفاعلى, يفعولا, وفعلاة, للفعلان, وفعلوان, افتعلناهم, افيعللي,
افعلالا, الفعللة, فعللوا, فعالللها, وفعللته, الفعلال, مفعللة, وفعل, الفعيلل, وفعللت, فعللت, والمفعلل, مفعلل, الفعاللة, تفعللا, والفعالل, والفعللة, فعللية,
وافعللت, افعللال, الفعلليلة, وفعللون, والفيعلول, فيعلول, الفيعلول, فعلليون, فعللي, المتفعلل, والفعاللة, الفعاللا, وفعللا, فعلولا, فعاللا, وافعلل, ومفعلل,
ويفعلل, فافعلل, يفعلل, افعللالا, بالفعلل, والفعلال, بفعلليها, والفعللي, الفعللي, وفعللان, وفعاللا, وفعاللة, وفعللي, فعيللان, فعيللانة, مفعللا, والفعلليلة,
فعللتها, متفعللا, الفعللا, التفعلل, وفعلال, وفعللها, فعلال, كالفعلل, افعللت, كفعللات, فتعللت, فعاللي, الفعللان, فعالله, تفعللت, فعلولها, والفعللان,
فعلليات, الفعللية, وتفعلل, فعلليا, فعلالة, الفعلالة, والفعللية, فعللتهم, وفعللتها, فعاليلها, فعوللة, الفعولل, فعللون, وفعالل, والفعللاه, الفعللوه, الفعللوت,
للفعللل, والفعللل, فعللا, فعاليله, فعلوله, فعلالها, فعلليل, الفعلليل, والفعلليل, والفعللى, بفعللا, فعللاة, فعللى, الفعللى, فعللل, الفعللل, مفعلله, لمفعلل,
فعلانون, والفاعله, وبالفعال, فعيلتها, والافعال, وفعلللت, الفعلللي, الفعلللانة, فعلللات, الفعلللى, فعلللة, الفعلللة, افعللل, وفعلللات, فعلللون, والفعلللة,
فاعله, فعلكة, والفعاليك, والفعليك, وفاعلني, تفتعله, فتفتعل, الفعيلا, المفعولين, وافواعل, وفعلوها, فواعيل, فوعلت, الفعلتين, والفعلتان, بفعوال,
افتعلنا, وافعلته, فعاليه, وفيعال, وفاعلا, بالمفعال, مفاعله, ويفتعلوا, فوعلتا, فعلتيهم, وفعلتا, وفعالها, للفعلة, مفعلها, بفاعول, الفعالان, فعيلى, وفاعلناه,
فعلاها, افتعالك, المفعلي, مفعلي, والمفعلي, وبمفعله, بفعيلة, بمستفعل, يفعلهم, والفعلية, تفعيلي, الفاعلات, للفاعلة, كالفعالل, فعلتي, وتفتعل, وتفاعلنا,
فعالتك, فاعلك, وتفعلتهم, المفعولا, وافتعليل, فوعلته, تفعليا, فالفعلة, بفعيل, وفاعلوه, فعلوه, لفيعل, تفعلن, بفعالها, للمفتعل, فعلوهم, التفعال, تفعلونا,
والافعيل, والمفعلان, منفعلة, كتفعيلك, منفعلا, المفعلين, ونفعل, فيفعلكم, وفعلناهم, فعلتنا, الفاعله, وافعالت, والمفاعيل, فعلناهم, استفعله, لمفعال,
وفيعلان, كالفعالة, ففعلان, وكفعل, الفعالات, فالمفعل, للمفعول, بفعلان, بفاعله, الفعيلان, والفعيلان, الفياعل, التفاعيل, والمستفعل, وفيعلون, وفيعلين,
الفعليت, وفعالاك, يتفاعلون, وافعلى, مفوعلا, كفاعل, لافعليين, كالمفاعلة, تفاعلا, والفيعلون, كالتفاعل, وافتعلل, والمفعيل, فعالت, والفعايل, والتفعال,
يفعال, الفعلن, مستفعلات, لاستفعاله, والتفاعيل, يفاعلون, بفعلين, وفاعيل, وفعلناها, الفعلتان, افعالت, تستفعلوا, والمتفاعل, متفاعلة, تفاعلت, والفعاله,
الفعللوة, وفعلله, فعلالا, الفعيلية, وتفعلة, يفتعلا, استفعلا, والفعالية, والفتعالل, ليفتعل, بفعلاة, بافعالين, تفعلة, وتفعلي, الاستفعال, والفعالون, فالمفعولة,
مفعيلة, افعيالا, افعيلت, والفعيلى, والفعلالة, والمفعللي, افعللى, فتفعلل, الفعللاة, وفعلالة, بالفعاليل, فعللته, الفيعلال, ليتفعلل, الفعللول, وفعللوات,
فاعلين, تفعلون, وفاعلتهم, بالفاعلة, مفتعلة, والتفعلة, مستفعلة, فاعلناه, متفعله, افوعل, بفاعلات, وفيعلا, الفعولات, يتفيعل, متفيعل, تفعولا, يتفعول,
وافتعاله, فافتعلت, فواعله, ومفعوله, بفيعلا, والفيعلانة, متفاعل, بالفعلتين, وفعيلى, الفعيلى, يفعلونني, المفاعلين, بالمفاعلين, مفعولي, مستفعلا, فعالهم,
فافعلوا, والفعولا, فتنفعل, افتعلها, ففعلهم, تفيعلا, وتفيعل, فاعيتل, وفعالي, المفاعلا, والمفاعلي, فيعيل, بمفعلات, بفياعل, بفيعل, فعاليا, والفعلولية,
الفعلوي, وفعلانيه, وفعليهم, فعلانيهم, بتفعال, افعولل, وفعلونا, فعلاوية, فعلاوي, فيفعلون, والفوعلات, وفعولي, الفعلو, الفياعله, والفياعلة, يتفعلوا,
الفعولي, تفاعله, بالتفاعل, ونستفعل, فاعتلاله, فافتعلتها, وفعلتنا, ووافعلتياه, ووافعلتاه, فعيللة, وفعلوت, والفعالنة, وبنوفعال, لفاعلة, ومفتعله, ليفعله,
وافعلني, لفعيل, بالمفاعيل, منفعيل, الفعلله, فعللانة, الفعللتان, يفعللان, الفاعلل, وفاعلل, المفعللا, لتتفعلل, وبفعلل, الافعللال, ولفعلة, مفعولان, ففيعل,
والافعلال, وفياعيل, لفعلانه, وتفعلها, فعيلاه, وتفاعلتني, ويفعله, فعلتن, وتفعيلها, استفعلوه, وافتعلوه, وفيعليل, والفاعلات, بمفعلة, الانفعال, فاعال,
بافعل, الفعيلي, وفعلناه, لفعله, والفاعلون, وفعليا, فعليت, والفعليت, يفعلوه, والمفعلون, فاعلتي, كالفعول, الفعالون, وفاعلنا, نتفعل, وتفتعلون, ولافعيل,
وفعالتها, لفعلن, ومفتعلات, ففعلني, لفعيلى, افتعالة, فافعلا, افعولت, فعالون, الفاعيل, متفاعلا, وفعالتك, وفاعلي, وفعالتي, ومفعولا, فيعلانة, والفيعلان,
بالفعالات, فعلاتها, للمفعل, وفعالته, ومسفوعل, مفوعل, وافعله, بالمفعالين, فاعلتنا, فاعلنا, مفعولون, وفعيلل, الفعيللات, مفاعلك, ومفعلون, وبفعل,
تستفعل, فعلتكم, فعالك, مفاعلته, تفعلتها, افعتال, وفعالهم, والمفتعلة, ومفعلها, يفعول, الفواعيل, استفعلوا, كمفعل, لفعلوا, فويعلان, فعالتها, واستفعلناها,
المفعللة, الفوعلل, والفوعلل, تفعيلك, وفاعلتها, يستفعلوا, فيستفعله, وافتعالها, ومفتعلها, فعالن, الفعلوت, فعلتمو, فعلوت, الفاعلتان, مستفعلين, ومفاعيلها,
كفعلول, فعليلة, الفعللوا, ومفعللات, والافعللال, متفعلل, ويتفعلل, فعللاوات, فعلليلا, افعللوا, فعلتلا, الفعلتل, ومفعللة, وفعليل, الفعاللات, الفعاللي,
يتمفعلون, وبالمفعلة, للتفعيل, والفعلاوان, المفعولل, والفعاول, وافعلوا, افعلال, فياعلا, فيعلاتهم, وافعوال, فسيفعلون, الفعيليل, فاعلني, فعلولل, لفعاللة,
فليفعل, بالفعلال, يفعللوا, بفعللة, وتفعللت, وفعليل, فعللها, ففعللا, فاعلول, بالتفعل, والفيعلي, افعيل, والفعالين, فتعللها, المفعله, ومفعالين, مفيعلان,
فوعله, افعلته, ليفعلني, وانفعالي, للفعيل, كتفعيل, وفعالات, فافعل, افعلوها, فتفعلها, افتعلتك, والفوعلي, بفاعوله, تفعيلهم, لفعيلة, كالفعلات, فاعوله,
لفعالات, افاعتلال, ويتفعلونه, فعولتك, نفتعله, كالمفعلة, وبفعلها, وتفتعلها, تفعلها, والفويعلة, الفيعلا, تفاعيل, مفعلته, وفعلانه, ففعلنا, والفوعليل,
المتفاعل, واستفعله, فعلاته, فاعولة, افعلان, سيفعلني, والفعيال, فاعلان, مفاعيله, وفعيللة, والافعلان, والفعلاو, وفعلنيه, تفوعله, فاعلوها, ومفعيل,
ومفاعلة, مفاعلاتي, يستفعلون, فعلهما, ويفعلونه, فعلتم, والفعلاني, ففعلوا, بالفاعل, فاعولا, ويتفعلا, والفيعوال, وفاعول, فعيلتنا, متفاعلين, مفتعلين,
والفعلالية, الفعالله, تفعللها, يفعللون, فعاللين, الفعللات, فاعلل, فعللاني, بفعلته, وللمفاعل, وبالفعل, بالفعلين, بفعلتين, والفاعلى, مستفعلك, استفعال,
تفاعلنا, وافتعلنا, الفعيلين, استفعلني, متفاعلون, والفعيليل, كالفعللى, فعيللا, فعللانيا, مفعللات, فعلاليل, فعللين, الفعللاني, فعللانية, والفعللاني, كفعلالية,
تفعيله, نفعله, والتفعلى, المفعلى, المنفعل, الفعاييل, والفعلكة, فعلانها, ففعوله, تمفعل, مفعيلا, المفعولات, وفعلياوان, فعلييان, فعلييون, فعليان, فعليون,

بفعلتيه, للتفعل, وفاعوله, فالتفعيل, الفاعلية, ففعلناهم, فاعلناهم, فمفعل, ويفعلني, فليفعلن, فانفعلوا, وافعلها, وانفعلوا, لفعول, فعيول, والفعيول, فعولكم,
الفياعلة, افعولة, والفوعلان, فوعالة, والفوعال, الفعوالا, فعيال, وتفعلني, فعلليلة, فعليله, فعاليلي, بمفتعل, وتفعلوه, بالفعولة, الفاعلا, فعولى, ففاعلو,
كفعول, فافتعلها, وفويعل, فويعل, الفعله, افعيالها, والفعاليان, وفيعالة, الفيعال, واستفعلنهم, ومفعولها, لفعولهم, فاتفعل, للفاعول, والفعلو, يفعلانه,
الفيعلان, الفعالاة, المفيعل, كالفيعال, اليفعلل, فعلله, كفعالة, فعالتيه, يفتعلك, ففتعل, كفعلتي, الافعالة, والافعالتين, بالافعالة, فعلاله, يفعلول, وافيعللول,
ومتفعلة, فافتعلنا, الفويعلة, وفويعلات, فويعلون, ييفعل, فعالتكم, لفعللة, وفعتل, لافتعالها, بافتعالك, لمفعولة, وتفاعيل, بفعلي, يتفعله, فعلنه, فاعلوهم,
بالفعيلة, الافعيلال, الافعلال, يفتعلان, متفعلاتها, وفيعليات, لتفعيله, الفيعلي, وفعلونة, الفعلانية, فعلاتة, متفاعلان, وفعللة, فوعالا, واستفعالي,
والفواعيل, الفوعللات, وفوعلاني, وفوعلي, والفاعال, كمفعال, فعلانيه, يفيعل, تفيعل, فعللنا, فعللولة, فتعل, والفعولل, والفوعيل, فيعلي, يتفعلونها,
وافتعللته, والفيعلل, فعايل, الافتعلال, لتفاعل, بفوعل, الفاعلي, التفعلة, للمفاعل, مفاعلون, ففعلتكم, فتفعول, فعوايلات, الفعوايل, يفاعلونها, فعولك,
افعلن, وافتعلها, يفعلا, فاعليل, ويفعلليني, وافعلليته, ففعلله, بفعللين, فافعللت, الفعلولية, الفعالول, فعولل, الفعلولل, والفاع, الفاع, فاع, فعالاة, للمفعلات,
الفواعلة, وتفعيلك, وكالفعالة, وكالفاعل, استفعلناه, ولفعيل, وفاعلوا, ومفاعلاتهم, والمفاعلات, وتفاعلته, نفعلها, والفاعولي, يفعلونه, كالفعللي, فعلتا,
وافعلي, افعلي, يتفعلا, وتمفعلت, بالمفعيل, والتفعلي, وتفعالا, فعولوا, وتفاعللوا, فواعلهن, ويفتعله, تفاعلها, ويفتعلك, بالفعالى, فيعالا, فعلوات,
فعالان, فاعيل, مفعلتك, فعلتما, ففعلة, بافتعال, افعلنا, افعلني, الفاعال, بفاعلين, فالتفعل, وفوعلت, وفعلاه, فعتل, بفعلك, فيفتعلون, واستفعلوا, وافاعتلال,
الفاعة, وافعلاني, والفواعلة, ويتفعل, كالمفعلاني, المفعلاني, مفعلاني, ليفعلوهم, فتفاعلوا, تفعولة, والتفعول, افعلالها, فعلويها, والفعلوان, افعلتها,
بفاعتها, فاعه, والفاعة, التفييعا, وتفيعت, وفيعة, فيعنا, تفيع, فاعت, والفياعى, تفيعت, بالفياع, والفياع, فيع, الفيع, فيعوا, فعوا, وفاعة, فاعة, الفاعات,
ويفعللها, الفيعلاني, فاعللة, فاعولتها, الفاعلاتين, فاعلاة, والفاعلا, الفاعولة, كالمستفعل, كفعالي, ومتفعلل, فيعلى, فالفعيلة, تفويعا, فوعه, والفواع,
وفعيلها, لمفتعل, لتفتعله, فالفعلي, الفعلولي, ومفاعلهم, وفعاولة, الفعوال, فعاليلهن, فاعللا, فالاستفعال, استفعلي, استفعالا, ليفعلوك, فعيلين, بالمفاعلة,
فلتفتعل, والمفعولي, للفعالة, بفعليه, فعليك, فعللوت, بالمفعلات, لتفعلكم, افعلوه, لفعيله, وللفعيل, مفيعيل, تفعيلكم, فعيلو, وبفعالته, بفعلانه, لفعوله,
الفاعلان, الفعلاوات, لفعلتكم, فاعلتكم, فواعلهم, المتفعلين, فعللان, فعلواني, الفيعلانات, فعيلل, للفعيلين, فالفعولة, بالفعلان, فعلانهم, وفعلانهم, فاستفعل,
مفاعيلهم, مفعلتان, والفعاليات, بالفواعلي, وفواعيل, كالفعاليل, الفوعال, الفوعالي, فعيللات, فافعلهما, مفعلتهم, بالفيعال, مفيعلة, فاعلنيها, بمفعله,
بالفعللة, بالفعلول, الفعلولين, مفعالي, وتفعيلات, وفاعلتان, واستفعلني, لفعلهما, فليفعلها, والفعلوتي, افعلوني, مفاعلهم, المتفاعلون, المفيعيل, بفاعال,
لتفعلنها, المفعلية, فعيلتين, الفاعلولة, كالمتفعل, تتفاعل, ويفعلوا, فعلتكها, الفعلاني, فيعلون, الفيعلون, الفعلاتوني, المفاعلون, وفعليك, للمفعلة, فعلانك,
لتفعلي, وفيعلي, واستفعال, فيتفاعلان, فواعلات, فافعلاني, بفاعيل, الفعليلي, الفعالتين, فاعلتك, فعلالو, الفوعيل, نفعلك, بفعولي, المفاعلات, بفعالهم,
وافتعلوهم, للفعاليل, بالفعلية, بفعلال, وافاعل, مفعلتي, فتفعلي, لتفعلني, كالفيعل, فيعلانا, الفاعلون, استفعلكم, لنفعلنك, فتفاعلت, ويفعلك, وينفعل, مفعلهم,
وتفاعلها, وفعولية, تفعلوها, ولفعل, والمفعولية, فعللاوي, اليفعولي, واليفاعيل, مفيعل, ومفاعلي, وانفعالا, افتعالها, بالافتعال, فعيلتك, وفعولى, وافتعلهم,
استفعلنا, وفاعلين, فعولي, بتفعيلها, فاعتلل, فتعللا, وفيعول, الفعيلال, المنفعلة, للفاعلين, وفعلهما, التفعلي, كمفعول, ويتفاعل, الفعليلية, الفاعيلون,
فعلنت, وفعلنه, تنفعلان, الفعليلات, لفعلهن, الفعيال, تفعيلية, الفعوليون, فعلوكم, وفعللل, ومفعلين, المتفعلة, وافعليه, الفاعولات, فليتفعل, يتفعلها,
فافعلوه, تفعللوا, الفعيلاوان, الفاعيال, الفاعيالات, بتفاعيل, للاستفعال, بفعللته, فنفعله, لفعولته, بافاعتلال, افتعليلة, وافتعليلة, فيتعلل, فتفعلوهم, الفاعالة,
وبفعيلهم, ويفعلهم, فليفعلوا, ففاعلها, فعليانية, بالفعليان, الفعليان, مفاعليه, وفعيلاه, كفعلاني, لتمفعل, بمفعيل, ويتمفعل, نفاعلهم, فعلوك, ومفعلاني,
وفعولتهن, التفعيلية, بفعلتها, لفعلين, وفعلن, ستفعل, ولتفعل, يفعيل, مفعالكم, ففاعلوه, والمتفعلة, بالفاعول, فافعلن, فالتفاعل, فتفاعلا, فعلانهما, بمفعاله,
ويفاعل, وفعلويه, والفعيلية, وفعاليات, للمفعلين, بفواعل, بمفتعلها, لافتعال, فعاليين, الفعلانة, والفاوعل, فواعلة, وفعلتك, ففعلن, والفعيلل, والفاعولات,
والنفعل, وفعلانات, تفعول, وتفعول, والفيعالتان, ومنفعلة, ففعيلة, فيعوله, افتعلهم, فالفواعل, فاعلتهم, تستفعلها, فاعولي, وبفعلك, بفعوله, لفعلها,
الفعلنة, والفعللاة, لمفاعيل, والمفيعل, ففعلتهم, لانفعال, المتفاعلان, كالفاعل, بالفعللان, وفعلانية, وافتعلني, تفاعلهم, وبالمفتعل, وبالمفعال, واستفعلها,
امفعل, ففعالها, والفعليات, مفعالان, وافتعلا, والفيعلاني, فعلاتهن, لفعلته, فعلوي, وفعالين, الفاعلوان, فالمفاعلة, مفتعلان, فوعلات, وافتعلله, فعلنته,
مفتعله, مفتعلنا, تفعلونهم, ويفعلونها, تفعلوهن, وفعلوك, الفعللين, الفعاليين, فعاليان, افتعالات, فعيولة, ومفتعلة, والفعلانين, فعليهم, يتفعللون, فتفعلته,
وفعلنتها, ومفعلي, وتفاعلا, افعليه, فعيلتان, فعيليلة, تفعلتهم, وبمفعالها, وبفعله, بافتعل, تفعلنا, فاعتللا, فعالنة, فتعللت, فاعتللها, بمفاعله, مفعالها, تفاعلته,
فاعلتانات, فعوليا, كفعله, استفعلتم, فعللتان, وافعيل, فوعلي, ولفعلنا, وبفعال, فيعلات, الفعتلان, وفعلوي, لفاعلون, للمفاعلة, وفعليات, بفعولها, اليفعيل,
وفعلاله, وفعوالة, وفعوال, كفعلانك, بفيعلتنا, وفوعله, وفعللن, بالفاعلات, وافعلوهن, مفعليها, لفعالته, الفعليانة, والفوعول, تمفعلت, مفعيلات, مفيعيلة,
وفاعولى, فعلوها, وفعيلتهم, بفعلاي, مفعلاة, ويفوعل, يفتعلهما, انفعاله, والافعولة, لتفعيلي, انفعلوا, وبفعلته, فيعلته, وفاعلان, وفاعلهم, والفعيلتان,
والافعل, تفعاليل, وفعلوكة, فعلوكة, وتفعلتني, فوعلية, والفعلاية, والفعلوك, وللفعل, والفعولان, بفعلى, ليتفعل, وافتعلوها, التفعول, وافتعلالة, لفوعلا,
ونفعلا, استفعلهم, واستفعلهم, لفعيلها, وفعيلون, وتفعال, والفعلاوات, فعولنة, لمفتعلون, وفعليته, بالافعال, ومفتعلون, وسناوفعلت, والفعولى, والافيعل,
الفعيله, واليفتعول, والفيعلانية, والفوعلانية, والفعلانات, فالمفعول, وفعلوة, والفيعلية, كالمفعال, مفتعلتان, ويفتعلون, ففاعلا, يتفعللن, الفعلانات,
والافعولته, والافعوال, وافعولت, والفعيولون, والفعاويل, والفعيولة, وتفعيله, لفعلات, بفعولته, الفيعالية, الفياعيل, والفياعل, افعولا, يفعلة, افتعللت,
فيفاعلك, مفعلتا, مفعلت, والفتعلل, الفعولين, فعليكما, وتفعللوا, الفعلاوة, المفتعلات, لفعلول, فعالكم, كالفعالي, فالفاعلة, وافاعلل, الافعوال, وتفعولته,
الفعاليتان, ومفعالا, لفعولية, والفعلنى, وفعلنى, وتفعلتم, وافتعلتهم, الفعتل, وفعليلة, مفعالين, يفعلي, فانفتعلت, والمتفعلات, فعلوانة, متفاعلات, لتفعلن,
فعلاويها, والمفعلاة, وتفعيلة, وتفعالة, متفعلتان, وفعوالا, فعولت, فعاليون, وفتعلل, اليفعل, وفعلو, والفويعلان, فوعلوا, ففعلوها, الفعيللين, وفعالتهم,
ومفعولل, الافعالا, بفعلتهم, والمفوعل, كالمفعول, والفعاليتان, الفتعللتان, وفتعللة, وفعيال, فعالاك, فعلوانا, وفعلوانه, وافيعالل, وافيعللي, نفاعلكم,
والتفيعل, ومستفعلة, ومفاعله, وبفعلة, فعاليت, وفوعلان, وليفعل, والمفعولل, الفعولوهو, الفعليات, ومفعيلة, لفعالا, فعيليل, والفاعيل, كافتعال, فعاليكما,
مفعوللا, فتعلله, وفاعولي, والمفعلن, تفعليه, تفاعلني, فعلايون, الفعلاية, وفعاليله, تفيعيلا, بالفيعلة, وفياعيله, والفوعلى, لمفتعلا, لتفعلهم, فعلانكوهو,
الفيعلانيات, المتفيعلون, فليفعله, وفعيلوا, والفعللتان, فعاولة, وافعتلنا, وافعتل, فيعتلل, وتفعيلا, فتعالل, فتعاللة, بالفعيلية, فعاليك, والفاعيلة, فاعلاوات,
لفعولي, وفعليون, المفيعلة, وتفعلات, لتفعلت, والفيعيل, الفوعلي, لفعلتها, تتفيعل, والمتفاعلات, وتفعلنه, وفعولوا, تفعلهما, وفعيلاي, فعلاي, فعتله,
افعيلاله, وفعلوتي, فعلاك, الفعلوى, والفيعلى, لمفعله, افاعيل, ومفعالة, وفعليان, الفعلانين, للفعيلة, فعلونا, وفيعله, لفيعال, يفعيله, مفعولين, فعوليون,
الفعلانيات, فيعولا, المفعولون, وفتاعل, وفعلنته, يفيعلون, يافاعل, وفعلنية, فعلينة, ووافعليتاه, وافعلاه, كالمفعولة, وكالمفعولة, والمفوعلة, مفوعلة,
وفعلنة, والفعاولة, الفعلوتان, فعلتيها, فيعلتك, الفعيليا, وافعلنا, الفعالنة, فعلونة, الفيعلى, والفعلواني, وفيعلى, افعيلاتا, وافتعالت, والفعليان, فافعلت,
وننتفعل, نتمفعل, تفعلتموني, بفعيله, مفعليا, مفعلية, ليفعلي, وافتعلنى, والفعوللي, الفعلولان, وفيعاله, ففتعلة, والافعيلان, وافعلانية, لفتعلل, وفعلتان,
وتفوعل, والفعيليات, افتعلن, وفعلانك, والفعويلة, لتفعيلهم, مفعللة, فعللى, المفعلل, والمتفعلل, بفعلله, وفعللوا, فعواله, ومفيعلاناتها, المفيعلان,
واليفعلية, فاعلتان, لتفعله, ومفعلى, يفوعله, والمفعلى, وفعليلته, منفعلين, وتفعالها, لمفعيل, بفعيلتهم, فيفتعله, الفوعلية, والفعالينة, وفعيول, لفاعلين,
استفعلك, باستفعاله, بفواعله, وافتعلال, مفعولتين, ويتفيعل, بفواعلك, افعلها, فيعاللها, وينفعلا, الفعيلتين, مفعلى, لينفعل, وفاعلتي, فيعلل, فعلوته, للفعالي,
وتفعللته, والفتعلل, وفعلللون, الفعلويل, وافعللال, وفعلولا, فعلولان, وفعللاة, وفعللين, الفوعليل, وفعللوها, مفعولتان, فيفعلونها, وتتفعل, لمفاعلتهم,
افاعلوا, والفعيلات, بفعالين, ليفاعل, استفعالك, وتمفعل, فعييل, وفياعلة, لفعلاني, الفعلاللة, مفعلل, والفعللولات, وافعللت, والفعاللى, وافعللى, لمتفعل,
فعلتيه, والفعيلون, فعلاوون, فعلاو, فعلاوة, وفعلاتهم, للفعيلي, وفاعلناهم, يفعلونك, بفعيلته, مفعوتل, والمفعوتل, الفعاتلة, الفعتلي, بمفاعلة, ومفوعل,
فعالو, وفعلتني, لفعلال, والفعلللل, وتفعللنا, فعللوهم, ومفعللها, لفعالل, الفوعلال, كفعاله, وكفعيلة, والفاعولل, فافتعلته, ويتفعلها, لتفعالة, مفعلانة, فاعلو,
وافاعول, والتفعييل, لافعلنك, يفعلنكم, المتفعلات, وتفعلى, فالمفعله, والفعالاة, لبفعلة, وفعالتان, وفعيلان, بمنفعل, الفيعوال, فعوللا, وفعلنت, الفعاليه,

والينفعل, افتعلتها, افعلانية, وفعلتيه, وفعالتيه, ومفاعلك, الفعلنون, فعلنين, وتفعلله, وافعلليت, بفعاليله, وبفعلاله, وفعلوله, والفعللانة, لفعلل, الفعللال,
الفعلاللان يفعلل, تفوعل, وتفيعلت, فاعليكون, اليفعلة, وفاعلوهم, نستفعل, فعياله, والمفعلةتكون, وافتعلتنا, ففاعلون, ولافتعاله, وفعللول, الفيعالون,
الفوعلى, لفعيال, فعيلكم, الفعيللية, بمفيعل, الافعلة, فعولهن, والفيعالة, يفعلت, والتفعلت افعلة, الافعلين, وفعيلين, لتفاعلها, تفتعلان, تفتعلها, يفتعلونها,
الفتعللة, الفيعلية, كفاعلة, واليفاعل, فعالليون, بفعاللة, وفعوللة, لفوعلة, الفعالن, افتعلله, الفعيلتان, كفعلي, وافتعللت, لافتعاله, واافاعلة, افتعللنا, المفتعلل,
والفعالانة, بافتعلل, افعيلي, والفعويل, مفيعلا, مفيعلون, مفعلك, فعليلته, افعللاة, المفعللي, وافعللاه, الفعاله, بفوعله, وبفاعله, ففاعلوا, وتفيعلة, بافعله,
الفاعلى, والفعيللان, كافعال, تفعلين, فعيالة, وافعاللت, فعيلاك, الفاعولي, ففعلوه, مفعتل, لفعلاتها, وافعاله, والفوعالة, الفعالينة, واستفعلناه, متفعللة,
وفعللى, الفوعيلة, يتفعلى, بافعالي, مستفعله, لفعلانها, الفتعلل, وفعاالان, بالتفعلي, تفعلتا, تفعلية, لفيعلي, لفوعلاني, لفعيلهم, فعلاية, والمفعللية, بالفعليل,
لفاعلان, ويفعيل, التفعالة, لافعلة, واليفعولة, افعلليت, افتعلتهم, بفعيلتها, والفاعالات, الفعليا, وبالفعيل, والمفعولون, مفاعلين, تفعاله, وانفعلنا, ففعلل,
والتفعلوت, تفعلوت, بفعالتين, ويتفعلون, والفوعلية, بافتعليلة, وفوعال, افعوللوا, فتفعلى, مافعلة, النفعيل, كفعلولة, لفعيلي, وفعالاهما, الفعليين, والتفعيلة,
كالفاعلية, وفعلتة, فعيلتة, لمفعلا, يتفعلني, للفعلول, فعلولاه, سيفعلون, فعيلتى, ففعللانى, الفعاتيل, والافاعل, بفعالات, فافتعلوه, فتفعلهم, وفاعلنى,
الافعل, فيتفعل, يتفعلتون, وبالمفعللة, وفعلالا, الفعليللة, الفعيليات, الفوعلان, الافعيلل, وافاعلا, فعالياتها, الافاعيل, والفاعليات, الفاعليات, الافاعل,
والفعللال, وافوعل, افوعلالا, فاعلوك, وفعلتنى, فعلوانى, الفياعول, فاعيال, فيعالها, مفاعلتي, فيعولة, بفعاليلها, وبمفعل, والافاعلة, الافعلان, الافاعلة,
لمفاعلتها, فيفعلله, فعلياوان, وفعلياوين, فعلياوى, فعلياواى, فعليين, فعللات, وافعللا, بمفعلل, ولافعلوله, الفعوللان, فعوللان, وفعوللان, وفعيللان,
بتفعيلك, وبلفعلل, والفعللين, فعلللاة, فالفياعل, مفعلن, وفعاللي, فعلوانات, فعللانا, فعللتين, وفعولل, ومتفاعل, بالفواعيل, الافعول, والافاعيل, بالفعللول,
وفعيليل, تفتعلنى, مفعولها, بالفعولل, فعلتهما, بفعيول, الفعيللة, الفعيللان, وفعتال, افاعلال, وافعلالا, كمستفعل, افعلى, بالفوعلة, مفعلكم, وفوعلا,
بمفعلنا, فافعلي, الفتعللان, تفعلوني, الفعلت, ففعلتها, نفعال, وفعللية, تفاعلتم, ونفعله, فعاليلا, فعيالا, وفعيالة, تفعالها, كالفعلول, وفعلوهم, وبفعلي,
فوعليا, الفعالك, الفعلوك, وفعالك, والمفعولات, فاعلوه, والافعول, افعيلا, الفعاويلات, الفعلالة, ومفعلته, وفاعلتني, بفعلتي, واتفعل, اتفعلالا,
والاتفعلال, فاعليه, نفتعل, بالتفعال, والافعيلل, وفعاليا, الفعللتين, مفعالتان, بتفعلوتها, الفعاة, فوعى, فعوت, وفعيته, فعيا, فعا, يتفاعلن, الفعاليا, ويتفعتل,
اتفعلل, متفعلات, فعلنتها, مفعلنا, فعليانة, ونستفعلا, افتعالتها, وتفاعيله, واستفعالك, انفعالهم, فالفاعول, فتعليله, تفعلنيهم, تفاعلن, فعلنية, فعلنيه, فعوتل,
افعالها, الفواعلا, لفعلت, ففعليته, كالفعيله, وتفعلا, ومفاعلون, وافعولاه, يفعلته, ففعلوهم, وفعلتهما, ففعلناه, يستفعلك, يفعلونهم, فعالهما, فعلتاه, بفعلا,
بمفعاليه, فمفاعل, وتفاعلوه, وفواعلة, فمتفعل, فعيالها, فاعلتيه, فعالاتهم, يتفاعله, الفعاولة, فعاوله, فاستفعلني, كفعلات, مفعالات, يستفعلان, وفعاليه,
المفعلتين, فعيلاي, فنفعل, الفاعلينا, واستفعلتك, بفعليها, ونفتعل, بفعلتك, فعالهن, يستفعلن, لتتفاعل, ووافعلاه, ففعلي, وافعلا, فيفاعل, لنفعلنه, وفيعولي,
وفياعيلي, فياعيلك, يتفعلان, يفتعلن, بفعيلها, وفوعلوا, فعيليه, مفعلتها, فعلتموه, ومفتعلهم, مستفعلو, بمفاعلكم, ومفاعلكم, فاعلاه, فواعلهما, ويفاعلهم,
واستفعلنا, والنفعلت, تفاعلاه, بفواعيلها, بفيعالها, ويفاعلني, تفاعلتا, فعلاتك, ومفاعلها, فتفاعل, بفيعلانته, فياعيله, وفواعلهم, فاعلتيها, افتعلوه, فمفعلة,
بمفعولة, وتفاعلاه, يفاعلن, ففاعلت, تستفعلون, ويفعلونهم, فيعالتيه, فيعالات, بفاعلته, بفواعلها, بالفعيله, فيعولي, مفعالهم, والفعلك, وفعلتين, لفعاوله,
فافعله, بالفوعال, فالمفعلة, فتفاعلاه, وبمفاعلهم, افعاليل, ليستفعل, وفوعلية, واستفعلك, يفعلله, وفعولهم, يفاعلني, بمستفعلين, الفوعله, ويفاعيله,
فعلاوين, وافعلن, مفعلنة, ومفعولتها, وفعيلهما, افتعلني, فواعلي, فافتعلناه, فعلالى, ومفعولوهو, فاعلية, تفاعلون, ومفاعلات, وبفعول, نتفاعل, بالفوعل,
وتفاعلوني, لبالمفعال, فعلتاهما, وفعلتاه, بفعولا, للفعاله, مفعلانه, بمفاعلهم, والمفعلتين, وافاعله, المفعلانية, وفعالنة, لفعلهم, وفعولات, بفعالاها,
فعالياها, تفعلونه, وفعالتاته, فعلولك, وفعليها, فتفاعلوه, الفاعليه, تفعيلاته, بمفعولين, وفوعلته, وفعيلتها, تتفاعلان, يتفعلهم, ويتفاعلون, ولفعلان,
ومتفيعل, فواعلكم, وافتعللني, كفاعلته, وفعلاته, وفاعلوها, فعلاكما, ومفعالها, والفاعلتين, ففعلتني, مفعليه, فتفعلا, فاعلوني, ومفتعلي, ومتفاعلة, فوعلها,
ومفعلتهم, وفعيلتي, لفعلتهم, وفعاليها, فيعاله, فيعالي, فياعيلي, فاستفعله, يفعلكم, فاستفعلت, وفعيلتان, تفعيلنا, تفاعيلهم, فليفتعل, وفويعلي, وفعلتي,
وفاعلاه, وفاعلك, كالمفعلي, بفياعيله, ومستفعلا, بفعلاتها, فاعلاته, لاستفعال, بفعاليل, ومفتعلك, وافتعلتك, يفتعلني, يفتعلهم, فيعلين, ومفاعيله, لفاعله,
فعولان, فواعلك, فتفعيل, وتفاعلناه, مفاعلنا, وفعلاتها, بمفعلتيه, بمفعلته, استفعلتها, لمتفاعل, فاستفعلا, ففاعلتهم, ومتفعله, كالمفاعل, وفعولتك, ومفعلتي,
وفعلانيهم, فعلتهن, فعلاتكم, ومتفعلاته, مفعلتكم, تستفعلان, فافتعلهم, تفعلتنا, ويفتعلنه, ويتفعلنه, وبفعاله, فعالتين, فعالتا, نفعلهما, يتفاعلك, الفعاليت,
المفاعلية, يتفعلونه, وتفعلهما, متفعلين, والفعولات, فعلتموها, ولفعلك, بفعلكما, ويفعلة, اليفعلات, فعلهية, ومفعلهم, مفعوللة, وفعاللتهم, مفاعلهن, مفتعلك,
ومتفعلك, كالفياعل, وتفاعلني, فعلاهما, ففعلناهما, فاعللها, فعلناهن, وفعاليلهم, فتعللوا, فاعتلله, فعلاتي, بفعللها, استفعالهم, فتعاللوا, والفعلتين, فوعال,
كالفعيلة, ومفعلتها, ويفعلن, ومفعاله, بفعالى, بفيعول, وفعيليلة, اليفعلية, والتفعلية, التفعلية, بتفاعل, ومفعليه, وفعلانون, فعولاتها, ويفتعلهم, وفاعلهما,
تفاعيلها, لمنفعل, وبمفعولة, منفعله, متفعلون, وبالمفاعل, بالفعولاة, لمفاعل, لاستفعلت, والفعاتيل, تفوعلا, بفعاليه, ستفعله, ومتفعلا, ومتفاعلا, فوعولا,
لمفاعلته, وبفعلات, التفعلات, وافعلة, بالفوعلي, ففاعلته, وفيتعلل, مفعلهن, ومفاعلهن, ويفاعله, بمفعليها, وافوعلة, افاعل, ويفعلي, بمفعليه, لفعالها,
وتفعوله, فاعتلالها, نتفعله, ونفتعله, فعتال, الفتعالل, لنفعلن, فعلناك, فتعللات, وفتعللها, بالفعله, فعليتان, فعالينه, ويفتعلونها, فاعولها, فبالفعال, افتعالهن,
واستفعلوه, يستفعلونه, يفاعيلها, اليفعلي, استفعلتهم, فاعليها, المفاعله, يفعلو, للافتعال, وفواعلها, المفعلان, افتعالنا, الفوعول, بمفتعال, تفعلنه, يفاعلنا,
واستفعلوهم, تفتعلانه, التفاعلية, واستفعلتها, ولافتعلت, فافتعلني, بفاعلتي, فويعلة, افقتعله, يالفعلة, وفعلتم, وبفعلين, المستفعلين, فاستفعلها, وبفواعل,
افتعلاى, يستفعلها, وساتفعله, ومفاعلتك, وتفاعلوها, والمفاعلون, واستفعلا, لبفعله, لمستفعل, مفعولك, فافتعلاني, فعلناكم, الفاعلول, الفاعللان,
الفعللانية, ويالفعلة, ونفعلل, فعللتني, الفعلالل, كافتعل, ففعال, لانفعالها, وفعلانى, افعاللت, والنفعلة, استفعلوها, فعلاهم, وفعلاهم, كتفاعيل, الفيعولا,
ففيعلا, فاعولته, فعلاويون, وافعيلالا, افعيلال, الفعتلة, النفاعل, ليفتعلن, لفعللا, فعللك, كفعلل, والمفعللة, فاستفعلي, وفعللاني, فلتفعل, فياعلها, فعاللته,
والفعيلان, وليفعلن, كمفاعل, الفعلوكة, فتفعيلا, فيفعلان, لتفعلون, وبفعيل, ففعيلها, الفوعالة, فعيليها, كفعالل, ففعللوا, والفاعليل, فيعلانيا, كالمفعلل,
واستفعاله, فعلهن, والتفعيلا, فعليلها, فعلالكم, ليستفعلها, ويستفعلون, فعلانكم, نفعلهم, فافعال, اليتفعل, لفاعول, والفعتل, بيفعل, اسفعل, فعلينان, فعاتلة,
فاعولك, بفاعولي, فعويل, بالفيعول, مفعلوها, وفعاليهم, كالفعلال, باليفعلي, وفعيللات, ففاعيل, الفعلوتين, بالفعلوة, والفعلولان, تفعلنهم, فعلتموهم,
افاعلتم, كالفعاله, فعاللى, فعالليان, يفاعلنهن, بتفاعلهم, فوعلين, كالفعللة, افتعلهن, بفعولة, وفعلنانة, الينفعل, مفعلتين, مفاعيلها, بالفعالين, لفعولها,
وللفاعول, فالمتفعل, الفعالتان, والفاعولية, تفاعيلي, تفاعيله, فعلواي, للفعلية, وفيعلية, بفعلوله, وبفعاليله, بفعليله, فعللال, الفعاللية, وفعالهما, لفعاليل,
ومفعلتك, الفعلالي, ففاعلني, فعللوه, افعلهم, فتفعلوها, نتفاعله, المفتعلون, مفتعلها, لافتعالهم, انفعالها, بفواعيل, الفواعلات, المنفعيل, وفعاللتان,
بالمنفعيل, والفعالا, فتفعللت, بفعللاتها, بالفعاله, تفاعلك, يتفاعلوا, الفعيالة, يتفاعلونه, والفعيلانة, ومفعولات, بفعلانها, انفعلا, الفعلليات, فعللتا, تفعلينا,
فتفعلني, وللفعال, لافعيلاله, ففعلك, ففعلهما, وفويعلا, ليفعلوه, فعولتها, وانفعلتا, وفواعلهما, فعللولا, وبفعلانه, بلفعل, الفيعالل, تفعلوهم, اليفاعل,
اليفعليين, واليفعليين, فعلويه, يفاعلا, لفعلل, فعوللى, والفعوللى, الفعوللى, فوعلتي, وفاعولهما, بفيعلة, بفيعال, ويفعلكم, بفعاليلهم, الفعالت, وفاعلو,
بفعلتكم, فعولبين, فالمفتعل, بالفعلاة, لافعلال, كفعالته, فعاللهم, ينفعلوا, بفعلكم, فعلوهن, فتفعللوا, فعليانات, يفاعلوا, فالفعلات, بفعالان, بافتعاله, نفاعل,
فعللتي, وفعللتيه, بالمفعولة, فعيليته, ولنفعلنه, يفعلونا, بفعلالة, ومفعلن, ومفعلنة, فيفعلن, ففعللت, الفعللولى, والفعللولى, فعللول, الفعاليك, كفعلى, فعيلتهم,
فوعلتها, وفيعالا, المفوعل, يفعلنها, متفاعلنا, ففوعل, والمفعللا, فيفعلهم, ولفعله, والفعيلا, وفياعيلك, ففعلى, فعلتة, وفعولن, فعولن, فوعلان, وافتعلاه,
فعلتاها, المفاعيلا, ليفعلك, فالفعلى, فعيلال, ليفعلكم, ولتفعلكم, ولنفعلكم, لنفعلكم, ويفعلونا, بفوعالة, وفيعلته, كفعلتها, فويعلين, نفعول, ليفعلها, يفاعلهن,
كالفعالات, ينفعلان, للفعيلات, الفعيلون, بالفعيلات, الفعللا, بفاعلي, والفيعلولة, يفتعلونه, كفعلها, وافعلل, تستفعلي, بفيعلانها, ففعولها, كالافتعال,
بافعلال, تفعلاك, تفعلاني, ففعليل, انفعلن, والفعلوانية, فعلوانتك, الفعلوانة, فعالاني, يفتعلكم, وفعاليلها, الفعاليان, نفوعل, بالمفعلين, وبالمفعلين, بفوعلة,
وتفعلوها, الفيعللى, تفتعلنا, ويفتعلوها, تفتعلوا, ليفاعلن, المستفعلون, ولفعول, فاعلتين, فالفعللة, كالمفاعيل, فعللاه, والفعلالل, وتفعلون, فعيلاوان,
بفعالك, كالمفتعل, يفتعلونا, يفتعلوه, فعلليلات, فتفيعل, لفعلليل, فاعالي, وفعاويل, منفعلان, وانفعاله, الفعولانة, الفعلللل, للفعلين, ويفعلهما, وفعيلن,

,فعيلنا يفعليه ,ويفعلياه ,وتفاعلك ,سنستفعلهم ,المفعالا ,فعتلة ,فعلايه ,لفعللوا ,فوعول ,كفاعله ,وتفاعلتها ,وافاعلتها ,فوعللات ,ففعلتان ,كفعلله ,ففعللوها ,بالفعالل يفعيلا ,فمفعالا ,الفوعلين ,ليفاعله ,والفعللون يفعلونهن ,كالفواعل ,وبالفعليا ,والفعليا ,بتفعول ,بالتفعول ,ومفيعل ,وكمفعل ,كافعله ,للفعولية ,للتفعلة ,المفعالة ,فعيلتيها ,ويفعلات ,لمفعلان ,كالتفعيلة ,فعالاتنا ,فعالنا ,فعولتهم ,فيفعلوك ,ففعول ,فمفعول ,والفعلوتى ,فعلانيون ,كالفعلى ,الفعلاوان ,الفعليلا ,وتفعيلهم ,وفاعليه ,والفيوعل ,للمفعلى ,ممفعل ,النفعل ,بنفعل ,لتفعيل ,ففعالته ,وفويعلة ,كفعيلة ,التفعلى ,ففعيلان ,وفعاليتان ,ويفاعيل ,باليفاعيل ,للفعيلتين ,افعلون ,افتعلنها ,والمفاعلا ,ففعاله ,وللفعالة ,فتفتعله ,بمفعالة ,المتفاعلة ,وفاعولها ,بالفاعولة ,فاعليين ,فاعليتين ,بفيعلان ,ومفوعلة ,المفوعلة ,ففعالا ,والممفعل ,فممفعل ,الممفعل ,فعلكما ,فعيلاته ,فالفعلن ,المفعلوها ,بتفعلها ,افعللانا ,والافعللان ,مفعللين ,فافعللوا ,والفعلنية ,وفعالاها ,للفعلي ,تفعلتي ,والمفعلاني ,بفوعلا ,فعلياوون ,فعلياوي ,فعلياواي ,وافعللتا ,بفعولله ,كافعل ,ويفعلوها ,فعالاتها ,وتفعلتا ,الفيعليين ,ففعللها ,لتفعللنها ,لتفعللت ,بفعالتهم ,بفعالل ,تفعلناه يفعللك ,ليفعلونك ,فوعلى ,ويفعلنا ,وتتفاعل ,تفعلوه ,فعلينا ,وفعلتاها ,وفعالياها ,وبفعولته ,فيعلولا ,فعليتين ,الفعليتين ,الفعليتا ,بالفعيليات ,وفعلولته ,والافعلل ,ويتفعله ,التفعيلات ,كالتفعيلات ,والمفعيول ,فاعلتم ,الفعلالا ,فاعليني ,افتعللته ,والفعاللي ,وفعلولها ,ومفعللا ,بتفعيله ,افعللاه ,لفعليك ,ففواعله ,افتيعل ,وافتعللها ,فاعليا ,كالفعلوت ,الفعلولا ,وفعليتها ,وافتعلن ,كيفعلون ,مفتعيل ,المفيعلون ,والمفاعلين ,افتعللوني ,والفعلللي ,الفعلكة ,بالفاعليين ,بالفعللي ,وفاعالات ,وفاعال ,فعلالي ,فاعلن ,وفعالاه ,الفعلالون ,الفعلانا ,ففعاول ,المفعلتان ,تفعلتك ,بفوعال ,والفعللا ,ولفاعل ,فعاللات ,فافعلوني ,وتفعولت ,فالمفعولات ,مفعلاه ,فوعلاه ,فعاويله ,فعاويلات ,فعللالا ,افعللت ,فعويلا ,للتفاعيل ,الفعلالول ,يفتعللون ,فيعليا ,بالفعللية ,فعلليه ,ومتفعلين ,فافعليها ,ومستفعلات ,مفاعلان ,فتتفعل ,ومفعولهما ,يستفعلني ,فاعلاك ,ليفعلون ,وافعليها ,فالافتعال ,واليفعلان ,ففاعلهم ,مفاعلونا ,الفويعل ,الفعيللا ,وفعليله ,الفيعيل ,فعولنا ,فعليلي ,يستفعلونني ,ويستفعلونها ,فعلولي ,بفعيلتين ,بالمتفعل ,فافتعلن ,مفاعلتكم ,سيفتعل ,فاعلتاها ,بفعالكا ,ففعلتا ,بالمفعله ,الفيعولان ,فاعللل ,تفعلتم ,فياعول ,بالفياعيل ,والمفيعلة ,افعيلانا ,والمفتعلات ,المتفاعلات ,بمتفاعله ,فعتيل ,فعتلا ,وفعلاتة ,الفعليول ,للفعلتين ,وافاعلوا ,بمفعلكم ,سيفعله ,فعالليا ,تفعلتني ,افعللا ,تفعليها ,بفعيلات ,المفعلينا ,كالفعلي ,ففعلول ,المفتعلان ,بفعلينا ,فعلكا ,فيفعلونه ,افتعلك ,وفعلوني ,وبفعيلته ,مفعولاته ,فيعلها ,كالتفعال ,بالمفعلى ,بالتفعلل ,تفعللن ,الفعليك ,الفيعلين ,لانفعالهما ,ليفتعلنكم ,مفعللله ,افعللا ,بالفيعلاني ,فالفيعل ,فاعلاهم ,والفعلللى ,فعللان ,بالفياعلة ,بالفعللى ,ولافتعال ,نفعلا ,بتفعل ,الفعلليون ,فعيتلوها ,بفعاللي ,الفياعلا ,لفيعلانه ,كفعللة ,بالمفعللة ,فافعلنا ,وفعلليلة ,فعللايه ,تفعيلت ,بالفيعلان ,لفعلنا ,سيفعلها ,يستفعله ,بفعيلتكم ,بالفعيلين ,تفيعيل ,فتيعيل ,نفيعيل ,فوعالات ,فعيليا ,وافعللوا ,الفيعلة ,والفعاة ,فوعي ,وفعوي ,فعوي ,فعلانينا ,فعلانيا ,متفعلينا ,وفعلكم ,فافتعلا ,لفعولتهما ,كالتفعيل ,وتفعللها ,كيفاعيل ,بيفعول ,وافعلوها ,كالفاعلة ,يتفاعلونها ,يفاعلونهم ,ففاعلتم ,فوعلتم ,فعلاواه ,الفعللوتات ,وفعللوه ,فعلانان ,فيفعلوا ,الفوعلاني ,ومفعولى ,فيفتعلان ,وبفعالي ,فعوليتهم ,تستفعلني ,لفعالي ,للمفعلل ,وفعالتاها ,ومفعولك ,ومفعللا ,وفعولان ,بفعالتها ,المستفعلات ,فعوللانة ,وفعيللانة ,ففعالل ,والفعوللة ,والفعوالة ,كالفاعول ,سنفعل ,لتفعلوه ,وفعلتموهم ,والفيعالية ,وفوعلانة ,وفيعلانة ,وفيعلاني ,فوعالني ,فعالوك ,ففعلون ,فالفعلون ,مفتعلهم ,باليفعول ,فعليتة ,كفعليت ,وفعلنياتها ,الفعلناة ,المفاعلي ,مفاعليان ,لبفعل ,بفعيلى ,والفعيللتان ,بفعلويه ,الفعلويه ,والفعلويهان ,والفعلويهون ,فعللهما ,تفعلوننا ,ليفعلوني ,والفعولن ,تفعلله ,يتفعلكم ,فعولين ,والمفعلانيات ,وافعيلاله ,وافعلوه ,والمتفعول ,فالفعيلان ,الفعولاوات ,استفعلوهم ,وللفعلة ,وفعولاه ,وفعللولات ,الفعللولا ,اليفعولة ,يفعولها ,لمفعلته ,المفتعلين ,وافتعلتني ,بفعاليها ,لفعلنني ,منفعلها ,فعالتنا ,فوعلك ,كتفعيله ,فاعلتهما ,كالفعلين ,والفعلويل ,فافتعلون ,فعيلتهما ,فعللانه ,وفيعلول ,افعلات ,ففعلنها ,بفعاليهما ,والفعيللة ,استفعاله ,والفعاللا ,الفيعولة ,وكفعلي ,لفياعل ,الفعلتن ,فاعلاتها ,فيعليتي ,فالفعلان ,ومفيعلانا ,ومفيعلانها ,مفيعلانات ,فعليكم ,المفعلت ,افعوللت ,مفعوللينا ,فعاللك ,يفعلليني ,فالمفعال ,والتفعلول ,بالاستفعال ,ليفعلن ,ويتمفعلون ,المفعوللة ,بالفعالية ,بيفتعل ,فمفعلات ,فعالليها ,فالفعلية ,بالمنفعل ,وفعولتي ,وفعالله ,الفعالل ,وفعاللية ,وفعللهم ,ففعللتها ,وفعالاته ,الفعيليين ,فالمفاعلا ,وفعويلا ,فيعلتهم ,وفيعلاتهم ,تتفعله ,فيعيال ,تتفعلون ,فاعتللهم ,لمنفعلة ,ونفعلة ,ونفعال ,الفيعولل ,فاعللوا ,وفعللتم ,فعلالية ,فيعلله ,والفعلولا ,فعلالين ,الفعللالان ,فيعللها ,فتعتلله ,كفاعول ,الفاعولية ,ليفتعله ,بالفاعلية ,فعلنات ,والفعتال ,ولنفعلا ,فعيل ,تفعىلا ,فعلتلت ,الفاعولا ,ماتفعل ,المتفيعل ,فتتعلل ,كتفيعل ,امفعلل ,بالفياعل ,مفاعيلن ,وفاعلانن ,فاعليان ,وفعلاك ,فيعللون ,كالفعيلا ,كالفعىلا ,ليتفعلوا ,والمستفعلة ,والفاعلاني ,فعالاتي ,ففعللاني ,فعلانيتان ,بفعيلهم ,فالفعليات ,مفاعلكم ,والفعولاة ,الفعولاة ,وفعولك ,للمتفاعل ,فعلانين ,افعوله ,وتفعللهما ,الافعللل ,لافعلل ,وفعيلاك ,ولتفعلنه ,والفعيلاة ,وكالفعل ,تفعلكم ,المفعللين ,ومفتعلين ,وفعلتكه ,وافتعلتكه ,بالمفعلي ,سيفعلك ,كفواعل ,بانفعال ,فيفعلوها ,فعاليلك ,فتتفاعل ,وافتعلناك ,مفعلاتهم ,مفعلاته ,وفعاللها ,فعللاته ,فافعلوهم ,كافعللت ,افعللها ,فيفاعلونهم ,يفتعلنه ,فمفاعله ,بالمفعلل ,والفعللانية ,الفعولى ,والانفعلال ,تفتعلون ,بفعلللة ,وفعاللى ,فالمفعلات ,يستفعلا ,بفعاللنا ,بفعولك ,ففعيله ,كفعيله ,فيعوال ,الفعلالية ,فعلليلته ,واليفعيلة ,فعللو ,استفعلها ,ففاعلوهم ,فيفعلوه ,تفعلنها ,ليفعلاك ,بفعلت ,بافتعلت ,لتفعيلها ,والفعتلان ,افعلاتا ,فافتعلوهم ,وبالمفعل ,وفعالون ,لفعيلكم ,فعللهم ,لنفعل ,لفعلونا ,فعيللة ,فعيلللة ,وفعيللاة ,فعولهما ,بفعيال ,فيعولية ,لاتفعله ,والمفعالا ,فعيلتيه ,ففعاليل ,والمفاعله ,بالفعيول ,فعاليتهم ,فتفاعله ,كافتعلال ,وتفعالتي ,افتعيل ,فتعللنا ,ويفعليل ,ويفعلول ,ويفعلولي ,لفواعل ,الفعاليلا ,وافعلال ,تفتعلني ,فتعليل ,وتفعلهن ,كتفعالة ,افيعلل ,افتوعيل ,فعلتياه ,فاستفعلنا ,لفاعتلال ,بفعللتيه ,افتعللة ,وافعيلانها ,وفتعله ,يفعلوها ,بفعاليهم ,كفعلت ,تفاعلينا ,الفتعللل ,الفعلللال ,فاعولنا ,الفعولبين ,والفعيلاوان ,وفعلوهن ,واستفعلتم ,ويستفعلن ,فاستفعلتم ,فتفعلنا ,وبالفعلة ,بتفعيلهم ,فيفعلهما ,فتعاللا ,فاتفعلت ,اتفعلت ,والفاعلول ,افتعلوني ,وللفعلات ,فافعلها ,بفعيلين ,فعيليني ,فالفواعلا ,وتفتعله ,بمفاعلته ,الايفعلان ,سيفعل ,فاعلناكم ,لمفعليه ,يسفتعل ,افتعالهم ,المتفاعلين ,فاعلكم ,افعلونا ,افتعلونا ,ففاعلة ,وفعيلهم ,فعولتنا ,فعيلنك ,فليفاعلها ,يفعليون ,وسفعل ,يسفعل ,سفعل ,سفعلا ,كتفاعل ,فليتفاعل ,فتفاعلوها ,تفاعلوها ,الفعاليون ,فاعلتني ,تفاعلة ,وتفعلان ,لاستفعالهم ,والافعيال ,افعيال ,فعايلات ,وفعايلات ,ويفاعلات ,ففعايل ,يفاعلاوات ,ليفعال ,افتعلوها ,ويفعولة ,مستفعلان ,يفعلنك ,فتفاعلته ,لفعلكم ,فياعلهم ,لتفتعلنها ,افتعلتم ,بفعالتك ,وافعلاها ,فيستفعل ,تفتعلونها ,لتفتعلون ,وافتعالي ,كالفيعلان ,الفاعليل ,فاعليلة ,تفعيلاتهم ,تفعيلات ,بالتفعيلات ,ونفاعل ,فوعلون ,فعيلهن ,والافعلة ,وفعلاها ,مفعلونه ,فتفعلتني ,ففعلوهما ,ومفعولته ,فعلالانة ,متفعلي ,فعالكا ,بالفعالك ,وفعلكي ,الفعلواني ,بفعلانهم ,فاعلاتي ,التمفعل ,لتفتعل ,وفواعلات ,للفعلل ,فاعلناها ,فلتفعله ,لمفعلهم ,وفيعلها ,تفعللون ,فافتعلوها ,فتفعلوني ,الفيعالاني ,فعلانن ,فعللن ,فعلنها ,وفعللنان ,الفواعليين ,ومفيعلا ,بالمفيعلات ,ففيعلوا ,فيعلوا ,فعايلهم ,فعايلكم ,الفعايل ,تتفعلوا ,وليفعلنهم ,فعلالان ,فاعلوهن ,فعلاكم ,تفاعلوهن ,يافعلى ,فستفعل ,لمفعولون ,فعولتهن ,لفعولتهن ,لمفعلين ,بفاعليه ,لتفعلوا ,ولتفعلوا ,وليفعلوا ,افعلك ,افعلكما ,افعلوك ,افعلوهم ,افعلتني ,افعلتم ,فافعلني ,لافعلتم ,لافعلوك ,لافعلناكم ,نفعلكم ,وافعلتهم ,وافعلك ,وافعلون ,يفعلوكم ,مفعليهم ,لفاعلوا ,يافعال ,ليفاعلوكم ,يفاعلونك ,كالمفعلين ,ففعلناهن ,لفعلناه ,لنفعلها ,وفعلناكم ,ولنفعلك ,ولنفعله ,ونفعلهم ,ليفعلنكم ,وسيفعلها ,وفعولهما ,وتفاعلون ,تفعلونهما ,وسيفعل ,فافعلهم ,وافعلهم ,ففاعلناها ,وتفعلونه ,يفاعلكم ,يافعلة ,يافعلتنا ,يافعلتى ,فسيفعلهم ,لنفعلنهم ,وافعلوهم ,لمفعلون ,يفعلوك ,وسيفعلون ,فالفاعلات ,ولنفعل ,فتفعلوه ,لنفعلنكم ,لتفعلنا ,وتستفعلوا ,وتستفعلون ,ولنفعلنهم ,ويستفعلا ,يفعلاكم ,يفعلنكما ,فعلكن ,تفاعلوهم ,فعلتموني ,لافتعلتم ,ليستفعلنهم ,ويستفعلكم ,بفعالكم ,بفعلهن ,فعلناهما ,والمنفعلة ,سنفعلهم ,سيفعلهم ,ليفعلنهم ,ونفعلكم ,ستفعلونهن ,فستفعلون ,فللفعل ,كفعلكم ,لتفعلة ,لفعلى ,لفعولنا ,وليتفعل ,ونفعلك ,فبفعل ,فكفعل ,فلفعل ,بفعولكم ,بفعولنا ,بفعولهم ,نفعلن ,ويفعلا ,تفعلونها ,ففعلناك ,يتفاعلا ,لفعلناك ,افعلهما ,بالفعلن ,بفعلتنا ,للفعلن ,تفعلانه ,بفعلنا ,بفعالاتي ,مفعلوا ,وبالفعول ,وبفعوله ,وبفعولي ,ولنفعلن ,ولفعوله ,لتفعلوها ,وفعالكم ,فافعلون ,لبفعيل ,ليفعلنه ,بفعلهما ,فعاليلهم ,ويفاعلون ,مفتعيلا ,ولنفعلنكم ,مفتعللون ,يافاعلي ,ولتفعلن ,يفعلونكم ,والمفعلين ,فياعيلهم ,للفياعيل ,للفيعال ,بفعالاتهم ,فعالتهما ,ففعالة ,لفعالتنا ,بفاعلكم ,بفاعلهم ,والفعيلين ,والمتفعلين ,بمفعلي ,ليفعلنها ,استفعلوني ,فيفاعله ,مستفعلون ,والمستفعلين ,انفعلتم ,فعلتموهن ,ففعلوهن ,ومفعلك ,ليفعلهم ,لفعالنا ,لفعاله ,يافعالي ,تستفعلوه ,ويستفعلونك ,يستفعلونك ,ولفعال ,فتفعلونها ,فلفعلتهم ,لتفاعلوا ,ولتفعلنهم ,افتعلوكم ,افتعلتموهم ,يفتعلوكم ,وفاعلوهن ,وفعيلتكم ,يامفعل ,تفعلونهن ,فيتفعلون ,واستفعلون ,فاستفعلكم ,واستفعلوه ,واستفعلي ,ويستفعلوا ,ويستفعلونه ,ستفعلون ,لفعالين ,لنفعلهم ,ففعيلا ,لفعيلا ,لفعلتا ,لفعلتم ,لمفعولا ,تفاعلونهم ,فلفاعلوكم ,فليفاعل ,فاعلوكم ,يفاعلوكم ,يفاعلونكم ,فافعليه ,ليفعلونا ,افتعلتموها ,وليفتعلوا ,بمفعلهم ,فتتفعلوا ,فينفعلوا ,فعولكما ,لمنفعلون ,لفعولكم ,متفعلكم ,منفعلون ,وتفعلك ,وفعولهن ,ينفعلون ,بفعيلي ,للمتفعلين ,ففعلكم ,ففعلتموه ,فتفعلون ,وبفعلهم ,وللفاعلين ,بفعلاته ,لفعلاته ,وفيعللوا .,وليتفعلوا ,ويتفعلوا ,فافتعلوهن ,لفعلناهم ,ويفعلونكم ,وتفاعلتم ,ولتفاعلتم ,يفاعلتم ,يفاعلنك ,استفعلوكم ,لافتعل ,وبفعلت ,فلفعله ,فسيفعلونها ,يستفعلوه ,فافعلوهن

ملخص البحث:

إنّ استخراج الجذور عملية أساسية مهمة في غالبية تطبيقات اللغة العربية، مثل: أنظمة استرجاع المعلومات، وتحليل النصوص، وتصنيف النصوص، وأنظمة الإجابة عن الأسئلة، وضغط البيانات، والفهرسه، والتدقيق الإملائي، وتلخيص النصوص، والترجمة الآلية. وإن أي ضعفٍ في استخراج جذور الكلمات من شأنه أن يؤثر سلباً على الأداء في تلك التطبيقات.

تُحقق خوارزمية سُنبُل لاستخراج الجذور دقة عالية في الأداء وتقدّم تصنيفاً جديداً للأحرف العربية من شأنه أن يقلل كثيراً من الغموض المتعلق بالبادئات واللاحقات. وتُبين المقارنة والفحص للخوارزميات المتاحة لاستخراج الجذور في اللغة العربية أن تلك الخوارزميات لا تزال في حاجة إلى بعض التحسينات.

يعتمد استخراج الجذور في اللغة العربية على استخدام الاوزان، إذ تزداد دقة العملية بزيادة عدد الاوزان. في هذه الدراسة، يجري تحسين خوارزمية سُنبُل لاستخراج الجذور من خلال تحسين قواعدها وزيادة أوزانها. وتم استخدام 4320 وزن لاستخراج الجذور، وتعد هذه القائمة أطول قائمة أوزان تم استخراجها من مجموعة قواعد بيانات "ثلجي". كما تم فحص الخوارزمية الجديدة باستخدام تلك المجموعة من قواعد البيانات التي تحتوي على 720000 زوج من الجذور والكلمات. والجدير بالذكر أن مجموعة قواعد البيانات المذكورة إنّما بُنيت لفحص خوارزميات استخراج الجذور ومقارنتها.

لقد تمت مقارنة الخوارزمية الجديدة بخوارزمية سُنبُل لاستخراج الجذور، وكانت النتيجة أنّ خوارزمية سُنبُل حققت دقة بلغت 68%، في حين بلغت دقة الخوارزمية المحسَّنة الجديدة 92%.

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 04, No. 03, December 2018.

FEATURE PRUNING METHOD FOR HIDDEN MARKOV MODEL-BASED ANOMALY DETECTION: A COMPARISON OF PERFORMANCE

Sulaiman Alhaidari and Mohamed Zohdy

(Received: 10-Oct.-2018, Accepted: 10-Nov.-2018)

ABSTRACT

Selecting effective and significant features for Hidden Markov Model (HMM) is very important for detecting anomalies in databases. The goal of this research is to identify the most salient and important features in building HMM. In order to improve the performance of HMM, an approach of feature pruning is proposed. This approach is effective in detecting and classifying anomalies, very simple and easily implemented. Also, it is able to reduce computational complexity and time without compromising the model accuracy. In this work, the proposed approach is applied to NSL-KDD (the new version of KDD Cup 99), DDoS, IoTPOT and UNSW_NB15 data sets. Those data sets are used to perform a comparative study that involves full feature set and a subset of significant features. The experimental results show better performance in terms of efficiency and providing higher accuracy and lower false positive rate with reduced number of features, as well as eliminating irrelevant redundant or noisy features.

KEYWORDS

Anomaly detection; Feature pruning; Hidden Markov Model; NSL-KDD; DDoS; UNSW_NB15; IoTPOT.

1. INTRODUCTION

Increasing size of data and network traffic has made information security more complex and challenging than at any time in the past. Information security is intended to protect information (data) and information systems from any malicious activities and unauthorized access [1]-[9]. However, the increasing size of data maximizes number of computations and minimizes detection accuracy rates [9]. Therefore, in recent years, many researchers have worked on this problem and applied the feature pruning method on several data sets to improve the detection performance and obtain faster and more cost-effective results. Using a feature pruning method for machine learning is a way to solve this problem.

This research examines the full feature set on NSL-KDD, DDoS, IoTPOT and UNSW_NB15 to reduce the number of features and identify the most salient and significant features that give higher accuracy and lower false positive rate. Therefore, a subset of significant features in detecting anomalies can be obtained by using one of the most popular machine learning techniques, HMM. These significant features can then be used in building an HMM that can effectively achieve much better prediction and detection performance. The remainder of this paper is organized as follows: Following the introduction to our work in Section 1, Section 2 reviews HMM and its problems. Section 3 describes the approach, followed by Section 4 which illustrates the experiments of this study. At the end, conclusions are presented in Section 5.

2. HIDDEN MARKOV MODELS

HMMs have been extensively utilized in many applications, such as speech recognition, finance, computer vision and bioinformatics. Hidden Markov Models can be defined as a tool for representing different probability distributions over sequences of observed variables [1]. In HMM, the sequence of observations is $O = \{O_1, O_2, \ldots, O_T\}$, where O_T is one of the observations symbols and T is the number of observations in the sequence. This sequence that goes through over time is observable and it's

S. Alhaidari[*] and M. Zohdy are with School of Engineering and Computer Science, Oakland University, Rochester, MI 48309, U.S.A. Email: [*]salhaidari@oakland.edu.

generated by a stochastic process. In contrast, the sequence of states X= {X1, X2…XN}, where N the number of states in the model, is not visible to the observer; therefore, it's not directly observed in this process. In a Markov chain, the probability of each state at time t is predicted based only on the previous state at time t-1 as shown in Equation (1). Figure 1 shows the first-order Hidden Markov model.

$$P = (X_{t+1}|X_1,..X_N) = P(X_{t+1}|X_t) \tag{1}$$

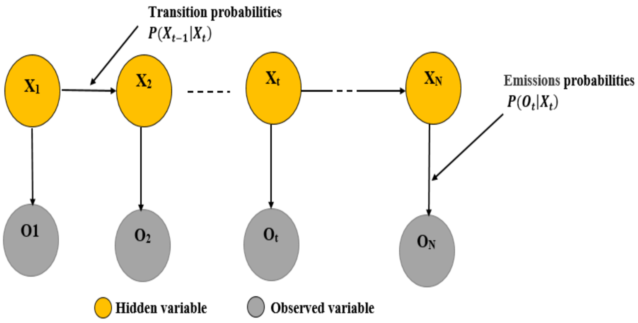


Figure 1. First-order hidden Markov model.

Mathematically, an HMM is defined as:

$$\lambda = (A, B, \pi) \tag{2}$$

where,
A is the state transition probability distribution. It's a single matrix N × N (N is the number of states), with each element a_{ij} representing the probability transitioning from state X_{t-1}, i to X_t,j as shown in Figure 2. It can be written as:

$$a_{ij} = P(X_{t-1}, j = 1|X_t, i = 1) \tag{3}$$

P is the emission probability. It is a single matrix N × M (N is the number of states and M is the number of emissions), where each element b_{kj} represents the probability of making observation O_t,k given state X_t,j as shown in Figure 2. It can be written as:

$$b_{Kj} = P(O_t = K|X_t = i) \tag{4}$$

π is the initial state distribution. I is an initial vector that contains the probabilities of starting in each of the states. It can be written as:

$$P(X_1|\pi) = \prod_{i=1}^{N} \pi_i \tag{5}$$

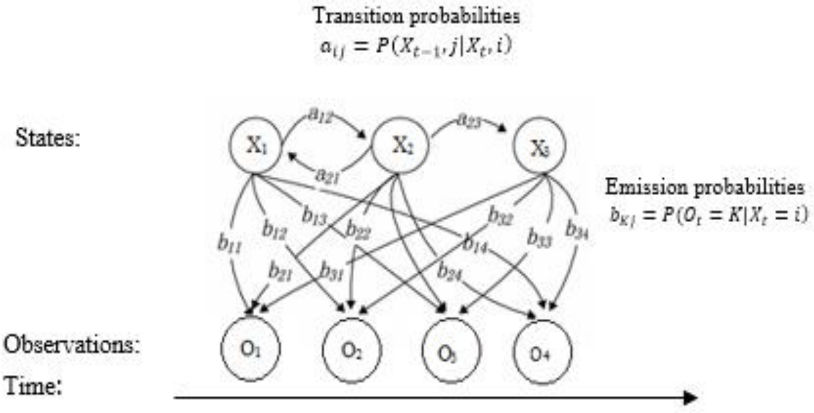


Figure 2. Structure of a hidden Markov model system, where the arrows between states X_1, X_2 and X_3 represent state transition probabilities and the arrows from states to emissions represent emission probabilities O_1,O_2,O_3,O_4.

Thus, an HMM can be characterized by a set of three parameters: the probability distribution of the initial states (π), the probability matrix of transition probabilities between states (A) and the probability matrix of emission probabilities for each state (B). HMM can be used in real-world applications by solving the following three fundamental problems:

- **The Evaluation Problem:** Given an HMM $\lambda = (A, B, \pi)$ and a sequence of observations $O = \{O_1, O_2, \dots, O_T\}$, compute the probability (likelihood)) of the observed sequence that was generated from the system given the model $P = (O|\lambda)$. This problem can be solved by two algorithms; namely, the forward algorithm and the backward algorithm.
- **The Decoding Problem:** Given an HMM $\lambda = (A, B, \pi)$ and a sequence of observations $O = \{O_1, O_2, \dots, O_T\}$, find the most likely hidden state sequence X_1, X_2,…X_N that produced the observed sequence. This problem is solved using Viterbi algorithm.
- **The Learning Problem:** Given an HMM **(**$\lambda = (A, B, \pi)$ and a sequence of observations $O = \{O_1, O_2, \dots, O_T\}$, adjust the model parameters (A, B, π) to maximize $P = (O|\lambda)$ to obtain the most descriptive model for the system. The Baum-Welch algorithm is used to solve this problem.

More expansion of these problems mentioned above and their solutions can be found in [2]

3. Feature Pruning Method

Feature pruning method is a method of eliminating features from the original dataset to obtain a subset of features. Reducing features from the full data set will not only reduce the data size that is required to process and the computational complexity, but selecting a good feature set also helps to improve the efficiency and accuracy of the detection model approach. It plays a key role in building detection models. On the other hand, using all features without applying feature pruning method might increase the overhead of the model, which leads to increase the time to build the model [7]. In this study, we present a feature pruning method which was built to be used to choose certain features from a given feature set. Our aim here is to find the significant feature set that gives the highest accuracy. In order to perform feature-pruning method, we first need to standardize the data to a normal distribution and then combine the standardized data to one sequence of observation, which then could then be used afterward with Viterbi algorithm to compute the most likely state sequence and then be compared to the actual sequence of states to determine the accuracy of the feature. The feature pruning algorithm that we implemented automates this process, eliminates each feature from the full set of the features and then checks the accuracy of the subset of features. More features that are least significant are eliminated if the obtained accuracy is within a certain tolerance of the accuracy, equal or higher than the previous accuracy of every feature combined. This process continues until no improvement of the accuracy is observed on elimination of features.

4. Experiments

4.1 Method of Performance Testing

To evaluate the performance of our proposed model and how accurate the model classifies and predicts the class label of attack and non-attack, we need to know the following four terms: True Positive (TP): the number of attack instances classified as attacks; True Negative (TN): the number of non-attack instances classified as non-attacks; False Negative (FN): the number of attack instances classified as non-attacks; False Positive (FP): the number of non-attack instances classified as attacks. For this study, we used the following performance measures to test the performance of the proposed model:

Accuracy: It is the ratio of the total number of correctly predicted instances to the total number of all instances. In our study, accuracy is measured by using the Viterbi algorithm to generate a likely state sequence and compare it to the known state sequence to get TP, FP, FN and TN. The accuracy can be calculated by using the following equation:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

Error rate: It is the ratio of the total number of misclassifications to the total number of all predictions.

Algorithm 1. Feature pruning algorithm for HMM.

```
Begin
feature_pruning (X, L, Already_Cheched)
Input: X-The number of desired features
       L-List of N feature Vectors
       Already_Cheched- A set containing already checked combinations of features
Output: R-The reduced feature set of length X
if Already_Cheched contains (L) then
    Return
end
if N-X=0 then //L contains the desired of features so test accuracy
    Return L, evaluate accuracy ()
else
    $A_i$= {All features in L except feature i} for all i=1,…, N //Create subsets of L with one less feature
    Return max (feature_pruning (X,$A_1$), feature_pruning (X,$A_2$), …, feature_pruning (X,$A_N$) //Return the subset with best accuracy
end
End
```

The error rate can be calculated by using the following equation:

$$Error\ rate = \frac{FP + FN}{TP + FP + TN + FN} \tag{7}$$

Fall-out: It is the ratio of the number of detected false positives to the total number of predictions. The fall-out can be calculated by using the following equation:

$$Fall\ out = \frac{FP}{FP + TN} \tag{8}$$

Sensitivity: It is the ratio of the total number of detected true positive instances that are correctly identified as attacks to the total number of positive instances. Sensitivity can be calculated by using the following equation:

$$\text{Sensitivity} = \frac{TP}{TP + FN} \tag{9}$$

Specificity: It is the ratio of the total number of detected true negative instances that are correctly identified as non-attacks to all the negative instances. Specificity can be calculated by using the following equation:

$$Specificity = \frac{TN}{TN + FP} \tag{10}$$

Precision and recall: Precision and recall measures are widely used for performance evaluation of machine-learning classification methods. Precision is the ratio of the total number of positive instances that are correctly identified as attacks to the total number of attacks. Recall is the ratio of the total number of instances that are correctly identified as attacks to the total number of all the instances that are correctly identified as attacks and misidentified attacks (it is the same as the sensitivity). Precision and recall can be calculated by using the following equations:

$$Precision = \frac{TP}{TP + FP} \tag{11}$$

$$Recall = \frac{TP}{TP + FN} \tag{12}$$

F-measure: F-measure is a testing score that tests the accuracy of the model and considers both precision and recall. F-measure can be calculated by using the following equation:

$$F - meaure = 2 * \frac{Precision * recall}{Precision + reacall} \tag{13}$$

4.2 Datasets and Discussion

4.2.1 NSL-KDD Dataset

The NSL-KDD 2009 dataset is a revision of the KDD'99 dataset that is extracted from the KDD'99 dataset to solve some of the inherent problems [3]. The size of NSL-KDD dataset is smaller than that of the original KDD'99. In each record of the set, there are 41 different features of the flow and one more attribute for class assigned to each record as either an attack type or a normal type as shown in Table 1. The NSL-KDD dataset contains 23 types of attack in the training set and 17 additional attack types in the testing set (New attacks that are not included in the training set) that are classified into four major categories: DoS, Probe, R2L and U2R, as shown in Table 2. The features in bold in Table 1 are the significant features obtained by the feature pruning algorithm used in our experiment on NSL-KDD dataset. Note: some features such as IP addresses, protocol type and source/destination port numbers were ignored from the initial feature set to guarantee that the results of the detection model are not dependent on particular acquisition biases. Table 2 summarizes the results of the experiment and Figure 3 shows the results in a graphical way.

Table 1. List of features of NSL-KDD dataset.

No.	Feature Name	No.	Feature Name
1	Duration	22	is_guest_login
2	protocol type	23	**count**
3	service	24	srv_count
4	**flag**	25	**serror_rate**
5	src_bytes	26	**srv_serror_rate**
6	**dst_bytes**	27	**rerror_rate**
7	land	28	srv_rerror_rate
8	wrong_fragment	29	**same_srv_rate**
9	urgent	30	diff_srv_rate
10	hot	31	srv_diff_host_rate
11	num_failed_logins	32	dst_host_count
12	logged_in	33	**dst_host_srv_count**
13	num_compromised	34	dst_host_same_srv_rate
14	root_shell	35	dst_host_diff_srv_rate
15	su_attempted	36	dst_host_same_src_port_rate
16	num_root	37	dst_host_srv_diff_host_rate
17	num_file_creations	38	**dst_host_serror_rate**
18	num_shells	39	**dst_host_srv_serror_rate**
19	num_access_files	40	dst_host_rerror_rate
20	num_outbound_cmds	41	dst_host_srv_rerror_rate
21	is_host_login	42	

Table 2. Summary of the experiment results for NSL_KDD dataset.

Measure	Training/Testing (80/20 %)	
	All features	**Obtained features**
accuracy	0.8431	0.8815
error rate	0.1569	0.1185
fall-out	0.1374	0.0477
sensitivity/ recall	0.8207	0.8009
specificity	0.8626	0.9523
precision	0.8388	0.9365
F-measure	0.8296	0.8634

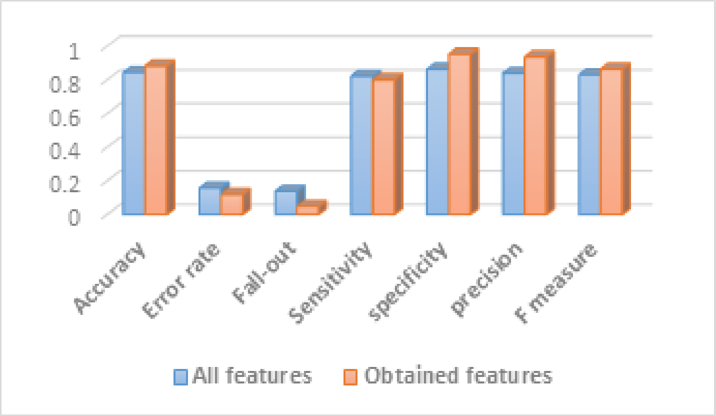


Figure 3. Comparison chart of the performance of all features and obtained features for NSL-KDD dataset.

4.2.2 DDoS Dataset

DDoS dataset is used for the evaluation [4]. It contains 27 features, which are labeled as either normal or an attack, as shown in Table 3. The DDoS dataset includes four types of the DDoS attack, which are: Smurf, UDP-Flood, HTTP-Flood and SIDDOS. From this dataset, a small portion of training and testing data is selected for experimentation. The features in bold in Table 3 are the significant features obtained by the feature pruning algorithm used in our experiment on DDoS dataset. Certain features were ignored from the initial feature set due to the same reason mentioned for the NSL-KDD dataset case. Table 4 summarizes the results of the experiment and Figure 4 shows the results in a graphical way.

Table 3. List of features of DDoS dataset.

No.	Feature Name	No.	Feature Name	No.	Feature Name	No.	Feature Name
1	src add	8	flags	15	pkt in	22	utilization
2	des add	9	fid	16	pktout	23	pkt delay
3	pkt id	10	**seq number**	17	pktr	24	pkt send time
4	from node	11	number of pkt	18	pkt delay node	25	pkt reseved time
5	to node	12	number of byte	19	**pktrate**	26	first pkt sent
6	pkt type	13	node name from	20	**byte rate**	27	**last pkt reseved**
7	**pkt size**	14	node name to	21	pkt avg size	28	

4.2.3 IoTPOT Dataset

IoTPOT dataset is Telnet IoT honeypot that analyzes malware attacks against various IoT devices, such as: IP Camera, DVR, Wireless Router, Customer Premises Equipment, Industrial Video Server, TV

Table 4. Summary of the experiment results for DDoS dataset.

Measure	**Training/Testing (80/20 %)**	
	All features	**Obtained features**
accuracy	0.8985	0.9741
error rate	0.1015	0.0259
fall-out	0.0098	0.0104
sensitivity/ recall	0.1071	0.8413
specificity	0.9902	0.9896
precision	0.5583	0.9038
F-measure	0.1797	0.8714

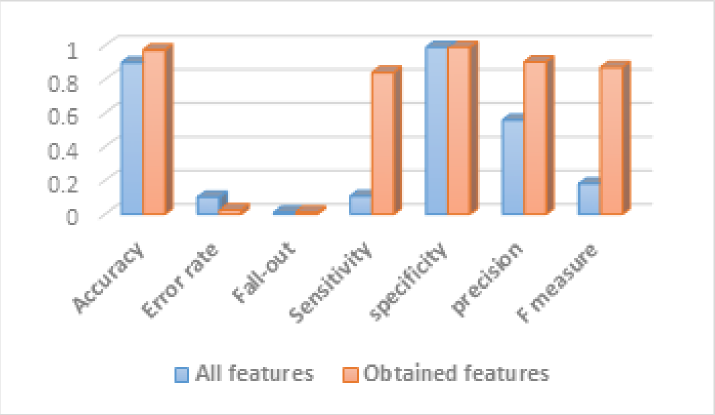


Figure 4. Comparison chart of the performance of all features and obtained features for DDoS dataset.

Receiver, Heat Pump, Environment Monitoring Unit (EMU system), Digital Video Scalar, home routers [4]. It analyzes the increase in Telnet-based attacks. This dataset contains 9 attack categories and 41 features, as shown in Table 5. The features are extracted by using NetMate tool set. Table 6 summarizes the results of the experiment and Figure 5 shows the results in a graphical way.

Table 5. List of features of IoTPOT dataset.

No.	Feature Name	No.	Feature Name	No.	Feature Name	No.	Feature Name
1	srcip	12	**sflow_fbytes**	23	std_biat	34	**min_biat**
2	srcport	13	**sflow_bpackets**	24	std_active	35	**mean_biat**
3	dstip	14	**sflow_bbytes**	25	**min_fpktl**	36	**max_biat**
4	dstport	15	fpsh_cnt	26	**mean_fpktl**	37	**duration**
5	**total_fpackets**	16	furg_cnt	27	**min_bpktl**	38	**min_active**
6	**total_fvolume**	17	bpsh_cnt	28	**mean_bpktl**	39	**mean_active**
7	**total_bpackets**	18	burg_cnt	29	**max_bpktl**	40	**max_active**
8	**total_bvolume**	19	**std_fpktl**	30	**max_fpktl**	41	**min_idle**
9	**total_fhlen**	20	**std_bpktl**	31	**min_fiat**	42	**mean_idle**
10	**total_bhlen**	21	std_fiat	32	**mean_fiat**	43	**max_idle**
11	**sflow_fpackets**	22	std_idle	33	**max_fiat**	44	Proto

4.2.4 UNSW_NB15 Dataset

The UNSW-NB 15 dataset was published in 2015 [5]. This dataset contains 9 attack categories and 48 features (shown in Table 7) which are categorized into 6 groups; namely: Flow Features, Basic Features, Content Features, Time Features, Additional Generated Features (General Purpose Features and Connection Features) and Labelled Features. Table 8 summarizes the results of the experiment and Figure 6 shows the results in a graphical way.

Figure 7 shows the ROC curves of the performance of HMM for the obtained features using the following datasets: NSL-KDD, DDoS, IoTPOT and UNSW_NB15. In terms of accuracy and the area

Table 6. Summary of the experiment results for IoTPOT dataset.

Measure	**Training/Testing (80/20 %)**	
	All features	**Obtained features**
accuracy	0.9222	0.9467
rrror rate	0.0778	0.0533
fall-out	0.0069	0.0188
sensitivity/ recall	0.1312	0.4786
specificity	0.9931	0.9812
precision	0.6316	0.6518
F-measure	0.2173	0.5519

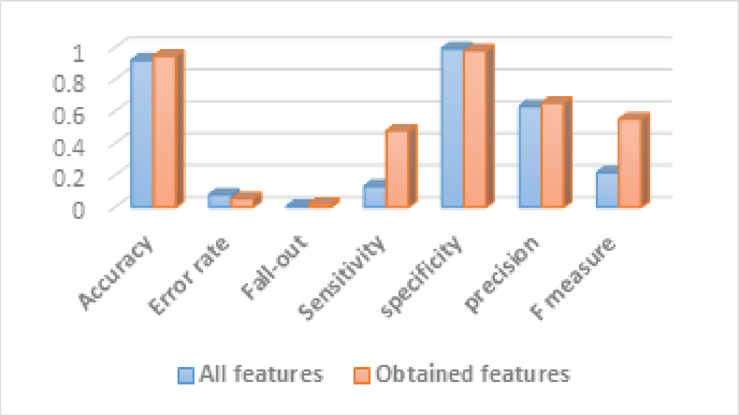


Figure 5. Comparison chart of the performance of all features and obtained features for IoTPOT dataset.

under the ROC curve (AUC), the DDoS dataset achieves the best results. The AUC estimates when the feature pruning method is applied to the datasets are shown in Figure 8.

Table 7. Summary of the experiment results for UNSW_NB15 dataset.

No.	**Feature Name**	**No.**	**Feature Name**	**No.**	**Feature Name**	**No.**	**Feature Name**
1	srcip	13	**dloss**	25	**trans_depth**	37	**ct_state_ttl**
2	**sport**	14	service	26	res_bdy_len	38	**ct_flw_http_mthd**
3	dstip	15	sload	27	sjit	39	**is_ftp_login**
4	dsport	16	dload	28	djit	40	ct_ftp_cmd
5	proto	17	**spkts**	29	stime	41	ct_srv_src
6	state	18	dpkts	30	ltime	42	ct_srv_dst
7	dur	19	**swin**	31	sintpkt	43	ct_dst_ltm
8	**sbytes**	20	**dwin**	32	**dintpkt**	44	ct_src_ ltm
9	**dbytes**	21	**stcpb**	33	tcprtt	45	ct_src_dport_ltm
10	sttl	22	**dtcpb**	34	synack	46	ct_dst_sport_ltm
11	**dttl**	23	**smeans**	35	ackdat	47	ct_dst_src_ltm
12	sloss	24	dmeans	36	is_sm_ips_ports	48	attack_cat

Table 8. Summary of the experiment results for UNSW_NB15 dataset.

Measure	**Training/Testing (80/20 %)**	
	All features	**Obtained features**
accuracy	0.8303	0.9641
error rate	0.1697	0.0359
fall-out	0.1916	0.0038
sensitivity/ recall	0.8459	0.9414
specificity	0.8084	0.9962
precision	0.8616	0.9971
F-measure	0.8536	0.9685

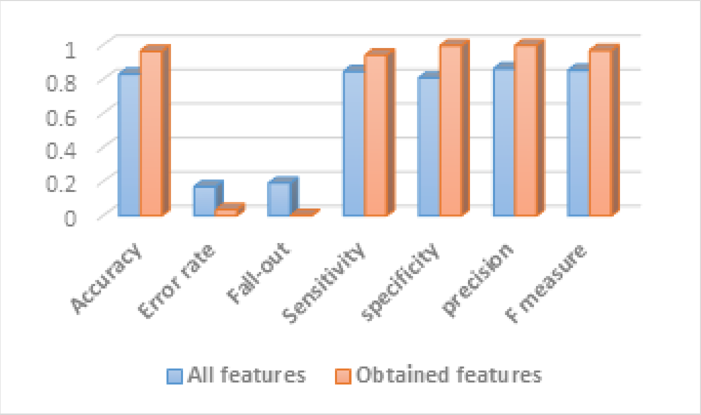


Figure 6. Comparison chart of the performance of all features and obtained features for UNSW_NB15 dataset.

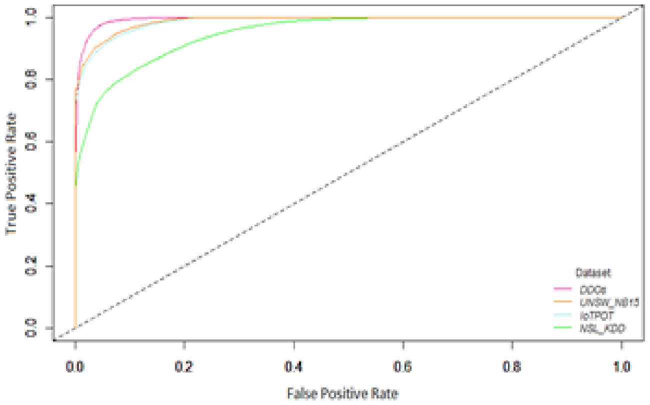


Figure 7. ROC curve of the HMM performance using the selected features on NSL-KDD, DDoS, IoTPOT and UNSW_NB15 datasets.

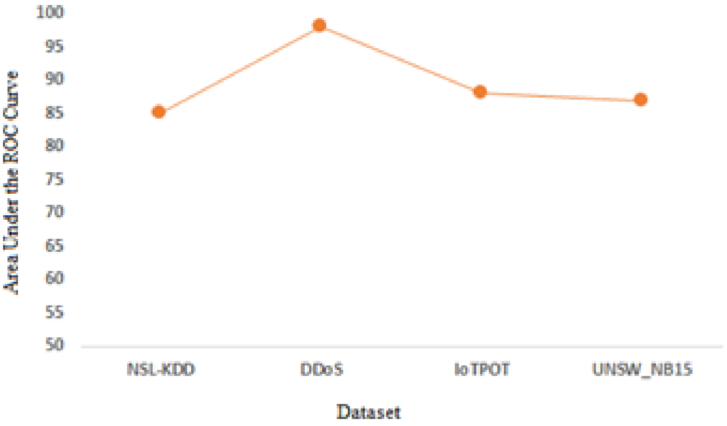


Figure 8. Performance (AUC) when feature-pruning method is applied on NSL-KDD, DDoS, IoTPOT, UNSW_NB15 datasets.

5. Conclusion

This paper proposed a feature pruning method for Hidden Markov Models to reduce the number of features and eliminate irrelevant, redundant or noisy features to overcome performance problems and improve the accuracy rate. In addition, this feature pruning method can effectively identify and determine the most significant feature set to be used for classification purposes. Experiment results were tested on four datasets: the NSL_KDD 2009, DDoS 2016, IoTPOT 2016 and UNSW_NB15 2015 datasets to show the superiority of our approach. The experiments demonstrated that the detection accuracy rate of using our approach is higher than the detection accuracy rate of full feature sets. In addition, false positive rate is lower than in full feature set. NSL_KDD 2009 produces 88.15% accuracy with 10 features. DDoS 2016 achieves 97.41% accuracy with 5 features. IoTPOT achieves 94.67% accuracy with 31 features. UNSW_NB15 achieves 96.41% accuracy with 16 features. As for the future work, we will focus on comparing the results derived from this study with other alternative machine learning methods. Meanwhile, we will continue to explore other datasets and flow-based features that could be used in order to improve the performance and achieve higher accuracy.

REFERENCES

[1] Z. Ghahramani, "An Introduction to Hidden Markov Models and Bayesian Networks," International Journal of Pattern Recognition and Artificial Intelligence, vol. 15, no. 1, pp. 9-42, 2001.

[2] Sulaiman Alhaidari, Ali Alharbi and Mohamed Zohdy, "Detecting Distributed Denial of Service Attacks Using Hidden Markov Models," International Journal of Computer Science Issues (IJCSI), vol. 15, no. 5, 2018.

[3] NSL-KDD Dataset, [online], Available: http://nsl.cs.unb.ca/nsl-kdd/.

[4] Pa, Yin Minn Pa et al., "Iotpot: A Novel Honeypot for Revealing Current IoT Threats," Journal of Information Processing, vol. 24, no. 3, pp. 522-533, 2016.

[5] N. Moustafa and J. Slay, "UNSW-NB15: A Comprehensive Dataset for Network Intrusion Detection Systems," Proc. of the IEEE Military Communications and Information Systems Conference (MilCIS), Australia, 2015.

[6] A. Alharbi, S. Alhaidari and M. Zohdy, "Denial-of-Service, Probing, User to Root (U2R) and Remote to User (R2L) Attack Detection Using Hidden Markov Models," International Journal of Computer and Information Technology, 2018 (Submitted).

[7] X. Zeng, Y.-W. Chen, C. Tao and D. Alphen, "Feature Selection Using Recursive Feature Elimination for Handwritten Digit Recognition," Proc. of the 5th International Conference on Intelligent Information Hiding and Multimedia Signal Processing, Kyoto, pp. 1205-1208, 2009.

[8] A. Alshammari et al., "Security Threats and Challenges in Cloud Computing," Proc. of the IEEE 4th International Conference on Cyber Security and Cloud Computing (CSCloud), NY, USA, pp. 46-51, 2017.

[9] A. Alharbi et al., "Sybil Attacks and Defenses in Internet of Things and Mobile Social Networks," CyberHunt 2018: IEEE International Workshop on Big Data Analytics for Cyber Threat Hunting, Westin Seattle, WA, USA, 2018 (Submitted).

ملخص البحث:

إنّ انتقاء مزايا فعّالة وذات مغزى لنموذج ماركوف المخفيّ أمر ذو أهمية كبيرة لكشف الشذوذ في قواعد البيانات. يهدف هذا البحث إلى تحديد أبرز المزايا وأهمها في بناء نموذج ماركوف المخفيّ. ومن أجل تحسين أداء نموذج ماركوف المخفيّ، يقترح هذا البحث طريقة لتشذيب المزايا. والطريقة المقترحة فعالة في كشف أوجه الشذوذ وتصنيفها، وهي إلى جانب ذلك بسيطة جداً وسهلة التطبيق. ومن ناحية أخرى، فهي قادرة على خفض التعقيد الحسابي وتقليل الوقت دون الإضرار بدقة النموذج. في هذا العمل، يتم تطبيق الطريقة المقترحة على أربعٍ من مجموعات البيانات (UNSW_NB15; IoTPOT; DDoS; NSL-KDD) التي جرى استخدامها لإجراء دراسة مقارنة تتضمن مجموعة البيانات ذات المزايا الكاملة ومجموعة البيانات الفرعية التي تشتمل على المزايا المهمة. وقد أسفرت النتائج التجريبية عن أداءٍ أفضل من حيث الفعالية، ودقة أعلى، ومعدّل أقل للخطأ في الكشف عن المزايا، بالإضافة الى حذف المزايا المتكررة أو المشوشة.

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 04, No. 03, December 2018.

SEMI-QUANTITATIVE SECURITY RISK ASSESSMENT OF ROBOTIC SYSTEMS*

Anas AlMajali[1], Khalil M. Ahmad Yousef[1], Bassam J. Mohd[1], Waleed Dweik[2], Salah Abu Ghalyon[1] and Roa'a Hasan[1]

(Received: 28-Aug.-2018, Revised: 15-Oct.-2018 and 5-Nov.-2018, Accepted: 15-Nov.-2018)

ABSTRACT

Robots are becoming increasingly integrated in our daily lives, providing services in civilian, industrial and military applications. Many of those applications require robots to be remotely operated and controlled through communication channels. This makes the robotic system susceptible to a class of attacks targeting the connection between the controlling client and the robot, which can render the robot unavailable. The objective of our research is to identify, estimate and prioritize the risks associated with attacks targeting the availability of the robotic system. To achieve our objective, we perform an impact oriented semi-quantitative risk assessment of the loss of availability on the well-known PeopleBot™ mobile robot platform. We experimented with several well-known attacks that can target and affect the availability of the robot. To examine the cyber-physical impacts of the attacks on the robotic system, we setup a ten-goal test area and constructed a 2D map. The robot was programmed to tour the test area while being targeted by cyber-attacks. The physical impacts of the attacks are demonstrated in this paper. The results indicate that attacks can potentially lead to loss of availability which may result in serious cyber-physical consequences.

KEYWORDS

Cyber-physical security, Robot, Availability, Threats, Attacks, Vulnerability, Risk, Risk assessment, Mitigation, PeopleBot.

1. INTRODUCTION

Robots have been an essential part in many domains like industry, military, research and health [1]. In the Internet of Things (IoT) era, robots are gaining increasing interest to perform critical and non-critical tasks. For example, robots can be used to clean the house floor [2], which is a non-critical task. On the other hand, robots can be used to remotely perform surgical operations, which is considered critical to human life [3]-[4]. As any Cyber-Physical System (CPS), robotic systems are prone to cyber-physical attacks [5]-[6].

Many robotic applications require remote control and monitoring by an operator like Unmanned Aerial Vehicles (UAVs) [7]-[8]. Such applications require establishing a bidirectional communication path between the robot and the controller. The controller can send direct commands to the robot to perform specific functions (e.g. move forward). Based on the application, the controller receives data from the robot (i.e., sensory data), which may be used to make critical decisions, especially in medical and military applications. The communication between the robot and the controller can be wireless (e.g. WiFi - IEEE 802.11) or wired (e.g. Ethernet - IEEE 802.3), utilizing different standards and protocols [9]–[11].

Robotic systems are susceptible to cyber-physical attacks, especially the ones targeting the communication path between the robot and the controller [1]. Attacks on the communication path causing loss of availability are referred to as Denial of Service (DoS) attacks. DoS leads to the loss of the communication between the robot and the controller, as well as the loss of the control and monitoring services. It is important to mention that losing the monitoring and control abilities while performing

*This paper is an extension to our published paper "K. Ahmad Yousef, A. AlMajali, R. Hasan, W. Dweik and B. Mohd, "Security risk assessment of the PeopleBot mobile robot research platform," in 2017 International Conference on Electrical and Computing Technologies and Applications (ICECTA), pp. 1–5, 2017".

1. A. AlMajali, K. M. A. Yousef, B. J. Mohd, S. Abu Ghalyon and R. Hasan are with Department of Computer Engineering, The Hashemite University, Zarqa, Jordan. Emails: almajali@hu.edu.jo, khalil@hu.edu.jo, bassam@hu.edu.jo, salah.g.ghalyon@hu.edu.jo and roaa.mamoun@gmail.com

2. W. Dweik is with Department of Computer Engineering, The University of Jordan, Amman, Jordan. Email: w.dweik@ju.edu.jo

critical missions (e.g. surgical operations, defense and space missions) may result in undesirable consequences and harm human lives. This raises the flag and demonstrates the importance of performing risk assessment on robotic platforms, especially if they are responsible for critical missions.

Despite the fact that cyber-physical security is very important, it is usually overlooked. This is evident by the study performed by [12], which indicates that about 30% of the studied robots are accessible from the Internet posing a security threat. The same study also indicates that 76% of the surveyed members did not perform a professional cyber-security assessment on their infrastructure, while more than 50% of the surveyed members did not consider cyber-attacks as a realistic threat. This demonstrates the lack of awareness of the current security posture in the robotic system domain.

The main objective of this paper is to perform cyber-physical security risk assessment on the PeopleBot™ research mobile robot platform [13]. By performing this risk assessment, we raise the awareness of possible threats on the robotic platforms. The robotic system, which is used in this paper as a use-case to perform the risk assessment, consists of the following components (see Figure 1):

- The PeopleBot robot, which runs a server application on its on-board computer to which the client connects. The client then connects to the server (i.e., robot), issues commands and monitors the operation of the robot.
- The client, which is simply a computer machine (e.g. PC or laptop) that runs certain applications to communicate with the robot.
- The network communication medium, which connects the robot and the client. In our case, it is the WiFi access point.
- The attacker, which performs cyber-physical attacks to affect the availability of the robot.

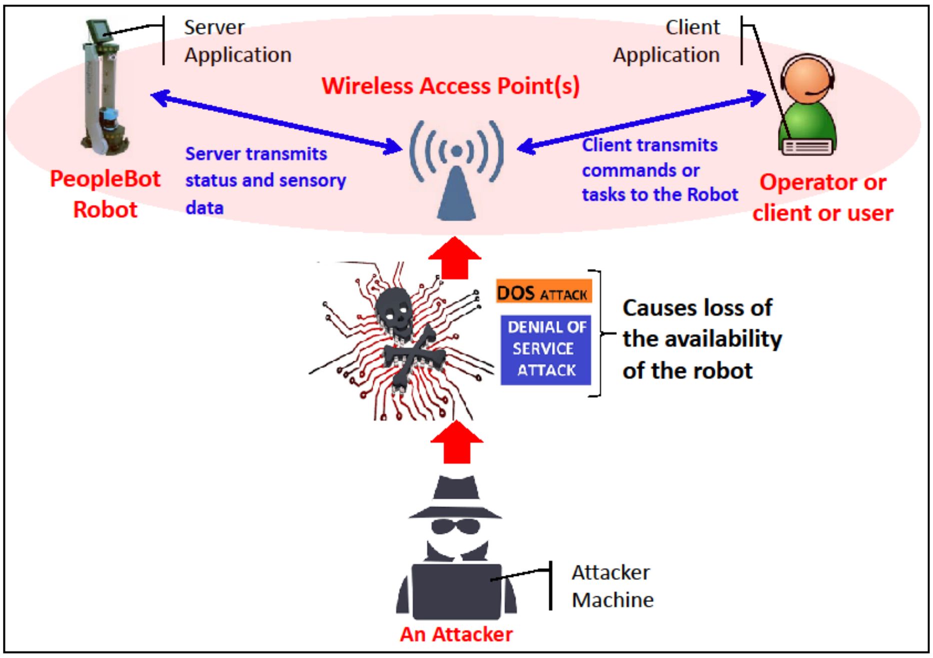

Server Application
Client Application
Wireless Access Point(s)
Server transmits status and sensory data
Client transmits commands or tasks to the Robot
PeopleBot Robot
Operator or client or user
DOS ATTACK
DENIAL OF SERVICE ATTACK
Causes loss of the availability of the robot
Attacker Machine
An Attacker

Figure 1. The PeopleBot robotic system under study.

This paper extends our previous work in which we performed qualitative risk assessment of mobile robots [14]. The main contributions of this paper can be summarized as follows:

- We identify possible threats and vulnerabilities that may lead to the loss of availability in the robotic platform using an impact-oriented approach. While those threats and vulnerabilities are not unique to the robotic platform and may apply to any computer network, it is important to study the physical consequences on robotic platforms and the impact on their critical missions.
- We create an experimental test area using the PeopleBot robot. We construct a 2D line and point

map of our research lab environment, identify and localize ten goals within the map for the robot to navigate and tour. We analyze the impact of the attacks on the robot while performing the experimental task of touring the goals. The goals are setup to emulate robotic tasks in a real environment.

- We estimate the risk of each identified threat by following a semi-quantitative approach based on NIST adversarial risk assessment template proposed in [15]. This assessment template was employed in numerous research work, such as the work in [16]–[18]. In this approach, representative numbers are used to estimate the risk of the loss of availability. The main advantage of this approach is that it compromises a middle ground between qualitative and quantitative risk assessments. To the best of our knowledge, this is the first study to perform such a semi-quantitative analysis on vulnerabilities in the robotic platforms.
- We prioritize the risk of each threat based on the risk estimation, so that more severe threats are handled first.
- We discuss the possible physical consequences of the identified threats.

The rest of the paper is organized as follows. In Section 2, related work is discussed. In Section 3, we introduce the PeopleBot mobile robot platform. In Section 4, we present the risk assessment approach used in this paper and the experimental results. Section 5 presents our discussion on the performed security attacks. Finally, we conclude the paper and provide our future directions in Section 6.

2. RELATED WORK

Recently, the security risk assessment on robotic platforms or CPSs has been a very hot research topic. Despite it is a critical issue, the number of publications addressing risk assessment on robotic platforms is limited. In fact, no systemic analysis of industrial robot security was conducted [12]. Further, there is inadequate understanding of what are the actual risks and the affected security goals [19]. In what follows, we present a literature summary for security threat analysis and detection for several robot platforms or CPSs.

Javaid et al. [7] analyzed different security threats to Unmanned Aerial Vehicles (UAVs). The threat model is based on listing vulnerabilities that can affect confidentiality, integrity and availability. The authors followed a typical risk evaluation approach.

Vasconcelos et al. [20] used three DoS attack tools to experimentally evaluate and analyze UAV behavior. The information gathering about the targeted network is performed using a reconnaissance attack that leverages an open source security tool called Network Mapper (Nmap). Next, DoS attacks are launched using Low Orbit Ion Cannon (LOIC), Netwox and Hping3 automated tools. All experiments were conducted in real time on AR.Drone 2.0 UAV while navigating inside a University building. The results show that Hping3 tool causes the highest average network latency of 455.82 milliseconds, which negatively impacts the video streaming application along with other computer vision applications.

Bezzo et al. [21] presented a control-level resiliency estimation technique against security sensor attacks in autonomous robots. The technique is based on a recursive algorithm, which exploits the redundancy in the sensor measurements. Although the proposed approach is generic, the authors used a case study on a vehicle cruise-control, where the latest N velocity measurements are recursively filtered and the variance of the measurements noise is considered to estimate resiliency. In addition, the authors validated the efficacy of their technique through outdoor experiments on two unmanned ground robots.

Bonaci et al. [19] discussed security threats for tele-operated surgical robot Raven II, which is an advanced surgery system. The authors demonstrated that intruders can maliciously control a wide range of the robot functions by performing disruption and manipulation attacks against the surgeon robot communication link that is likely to be wireless. The attacks are based on man-in-the-middle model and they successfully impacted the safety and usability of the surgical robot, which could potentially result in legal and privacy violations. Batson et al. [22] conducted an analysis to identify threats and vulnerabilities in the system’s concept for Unmanned Tactical Autonomous Control and Collaboration (UTACC). Jones and Straub [23] presented a two-stage intrusion detection system (IDS) for detecting network intrusions and malware in autonomous robots. The authors utilized and trained a deep neural network to detect commands that deviate from the expected behavior.

Maggi et al. [24] studied the impacts of system-specific attacks on real industrial robots. The authors analyzed two attacker models: network attacker (i.e., communicates with the robot over the network) and physical attacker (e.g. the robot operator). Five attack scenarios were demonstrated (e.g. altering the control-loop parameters, tampering with calibration parameters, …etc.) and the physical impacts in addition to the compromised requirements (i.e., safety, integrity and accuracy) were discussed. Similarly, Quarta et al. [12] performed an experimental security analysis of an ABB 6-axis IRB140/IRC5 industrial robot controller. The authors exploited several software vulnerabilities in the robot main computer (MC). They mainly exposed and focused on the network services that are essential for the operation of the robot, such as FTP (File Transfer Protocol).

Lera et al. [25] presented a taxonomy that classifies cyber-security attacks, which target safety and security of service robots. For safety threats, the proposed taxonomy differentiates between the risks according to the user type (e.g. domestic, commercial). For each user type, the expected risks are classified according to the level of the physical impact (e.g. destruction, partial damage) and the origin of the risk. On the other hand, security threats are classified according to the robot function (i.e., personal or commercial) and the type of sensors that the robot is equipped with.

Vuong et al. [26] proposed two different approaches for detecting attacks in robotic vehicles. The first approach was based on using decision trees and the second approach was based on using deep learning. Loukas et al. [27] argued that the limited rule-based or lightweight machine learning techniques used for cyber-physical intrusion detection of vehicles can be substituted with more advanced techniques using computational offloading to the cloud. The boosted processing power is used to implement a deep multilayer perception and recurrent neural network architecture, which receives the real-time cyber-physical data captured in the robotic vehicle and analyzes it to detect intrusions. The authors showed that the deep learning technique noticeably improves the detection accuracy; however, the long detection latency and the security of the external network between the vehicle and the cloud must be carefully considered.

Similar to this work, some researchers focused on risk assessment for robotic platforms. Chen et al. [28] assess the cyber security risks in industrial control systems (ICSs) (e.g. SCADA) by quantifying the availability using the concept of mean failure cost (MFC). Various security issues arise as the ICS becomes more integrated with IT networks. Hence, it is important to compare the cost of implementing security counter-measures with the expected losses of cyber-attacks, especially due to the limited resources.

Dominic et al. [29] proposed a risk assessment framework for autonomous and cooperative automated driving. The authors started by describing the recent attack surfaces and then discussing the proposed application-based threat enumeration and analysis framework. For each threat, model parameters are specified and accordingly the result vector which characterizes the risk level of the threat is computed. The result vector reflects attack potential, motivation and impact.

Very recently, the authors in [30] performed qualitative risk assessment of several vulnerabilities identified specifically to the Adept mobile robots (e.g. the PeopleBot [13]); namely, the MobileEyes/arnlServer client/server robotic applications. Such applications are necessary to establish the network connection between the Adebt robots and their clients or users. In contrast to this work, this paper proposes a semi-quantitative security risk assessment, where representative numbers are used to estimate the risk of the loss of availability. The main advantage of this approach is that it compromises a middle ground between qualitative and quantitative risk assessments. Additionally, the focus of this paper is mainly on identifying and raising awareness of possible threats and vulnerabilities that may lead to the loss of availability in the robotic platform under study, the PeopleBot.

3. THE PEOPLEBOT MOBILE ROBOT PLATFORM

Figure 2 shows the PeopleBot mobile robot from Adept company[13], which is used as a case study for our proposed security risk assessment. The PeopleBot robot is a research platform that can be used in service and human robot interaction (HRI) projects and in other projects as well [31]. It consists of multiple hardware and software components: main computer, mechanical actuators, controllers, sensors such as lidars (or laser range finders) and cameras, human-interaction devices, control logic, firmware and operating systems (either Windows 7 or Ubuntu 12.04). The main computer of the PeopleBot is connected to the sensors either through controllers or isolated subnet *via* an on-board network access point. There are various interfaces or ports on the robot that include-but are not limited to: serial port, Ethernet RJ45 port, USB ports and wireless adapter.

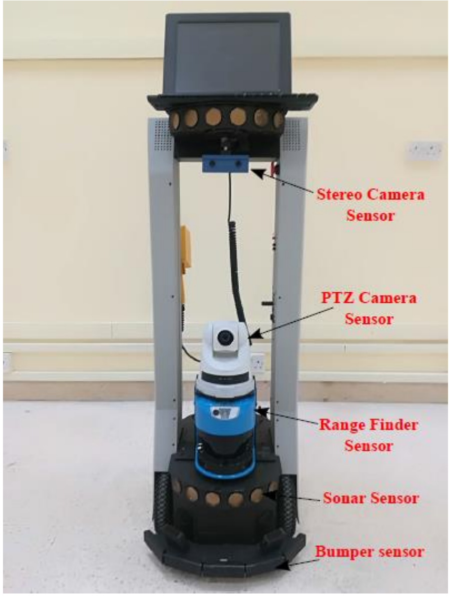


Figure 2. The PeopleBot mobile robot platform, where several sensors are attached to the robot (e.g. pan-tilt-zoom (PTZ) camera, laser range finder, sonars, stereo camera, …etc.).

Several software packages [13] are pre-installed on the robot main computer, such as ARIA, ARNL, MobileEyes, Mapper3, …etc., that enable the control of the robot, its sensors and accessories. In many applications, the robot is often required to be remotely accessible either through a connection to the Internet, or *via* dedicated wireless access points. Some of those applications include:

- Museum robotic guide application [32].
- Map building and robot self-localization [33]-[34].
- Robotic assistant for healthcare applications [35].
- Gesture-based multi-robot control application and robotic desk clerk application using face recognition [10].
- Application of Myo Armband System to control a robot interface [36].
- Extrinsic calibration of camera and 2D laser sensors without overlap [37].

As it can be seen, all of the above applications of the PeobleBot are very important, critical and thus must be protected against cyber-physical attacks. Consequently, this is one of the main goals of this research.

3.1 Experimental Setup

The experimental setup used to perform the impact-oriented risk assessment on the PeopleBot robotic system (shown in Figure 1) assumes the following components:

- The robot runs a server application on its on-board computer. Such an application is assumed to communicate and access all the sensors pre-installed on the robot, fully control the robot movement, have already established a network connection to a certain network and allow (single or multiple) connections from authorized clients to connect to the robot to access/ monitor all of its features, sensors and movement.
- An operator or a client runs certain applications to communicate with the server on his/her own computer.
- The robot network to represent a wireless access point, in which the client can connect to the

server in a client-server mode.

- Only cyber-attacks that can result in the loss of availability on the robot are considered. The attacks target one or more of the following:
  - Application resources; e.g. server application.
  - System resources; e.g. network handling software.
  - Network bandwidth.

To evaluate the impact of the attacks, we created an experimental test area using the PeopleBot robot. We created a 2D line and point map of our research lab environment (6 × 9 m^2 area). We identified and localized ten goals within the created map of our lab as shown in Figure 3 for the robot to navigate and tour. These goals may emulate important stops for the robot within an industrial workplace application or other applications. The goals are 60 cm spaced apart. The number of goals and separating distances were chosen based on initial feasibility trials to provide adequate accuracy. When a "tour goals" command is issued, the robot tours the goals one by one starting from goal 1. The robot sends its real-time coordinates to the client so that it monitors the physical location of the robot on the map.

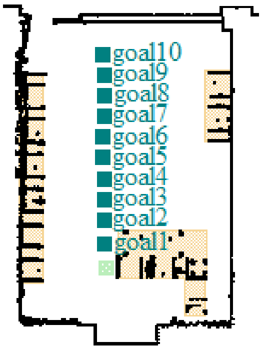


Figure 3. Ten identified and localized goals within the 2D line and point map of our research lab.

4. Risk Assessment of the PeopleBot Robot

Risk is usually defined as the expected impact of an event on a system or organization [15]. Within the context of this paper, an event refers to a cyber-attack and a system refers to the robotic system. Therefore, the risk determination is a function of the impact of an attack and the probability that this attack occurs and succeeds. The impact of an attack measures the loss of one or more of the main security requirements: confidentiality, integrity and availability. The main focus of this research is only on the availability security requirement. The probability that an attack occurs and succeeds depends on the vulnerabilities and threats of the system under study (i.e., the robot). Hence, the risk determination of a given attack can be expressed as follows:

$$Risk = Vulnerability \times Threat \times Impact \tag{1}$$

Risk assessment is the process of identifying, estimating and prioritizing risks to a system or organization [15]. The flow chart in Figure 4 is adopted from NIST risk assessments [15] and shows the main steps of the assessment process. Next, we present the details of the risk assessment process for the attacks under study.

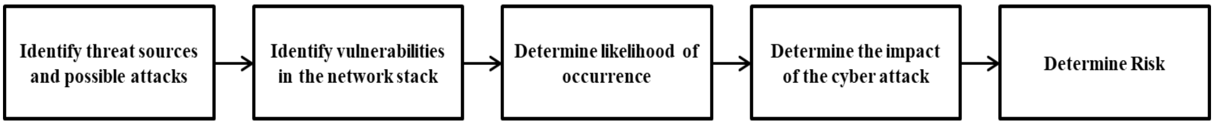

Identify threat sources and possible attacks
Identify vulnerabilities in the network stack
Determine likelihood of occurrence
Determine the impact of the cyber attack
Determine Risk

Figure 4. Risk assessment process [15].

4.1 Risk Identification

To identify the risk, we follow an impact-oriented analysis approach to assess the risk of the attacks on the robot. We focus on the loss of availability as the impact, because this could lead to devastating consequences. As mentioned earlier, the user can connect in a client-server mode to the robot at the application level to perform certain functions (e.g. transferring medical equipment) and retrieve data (e.g. geographical coordinates). The client and the server have to be available in order to perform the required functions of the robot. Losing availability may have physical impacts, like human injury or physical damage of the robot itself.

4.2 Risk Estimation

To estimate the risk, we have to assess the vulnerabilities and threats that may cause loss of availability. We follow a semi-quantitative risk assessment approach to estimate the risk. In this approach, representative numbers are used to estimate the level of each risk factor, like the severity of a certain vulnerability. The semi-quantitative assessment approach represents a middle-ground between qualitative and quantitative approaches. The semi-quantitative assessment facilitates better comparison and prioritization compared to the qualitative assessment approaches, which rely on non-numerical levels (e.g. low, moderate, high). On the other hand, the semi-quantitative assessment is easier to implement than the quantitative assessment, which requires using specific metrics to measure different risk factors in the assessment process (e.g. the impact of the attack) [15]. Identifying those metrics is challenging in the cyber security domain.

4.3 Risk Prioritization

After estimating the risk level for the attacks under study, the associated threats can be prioritized based on the numerical values calculated to each threat. Next, we present our risk assessment of the loss of availability of the robot.

4.4 Attack Analysis

Figure 5 demonstrates the attack tree for the loss of availability. The root of the attack tree represents the ultimate goal of the attacker, which is causing the loss of availability. In addition, the loss of availability may lead to cyber-physical threats depending on the task that is performed by the robot. The branches of the tree represent the ways and techniques through which the attacker can achieve the ultimate goal. There are three main ways to perform a Denial-of-Service (DoS) attack: attacking the network bandwidth, attacking system resources or attacking the application resources. The next level of the tree represents the techniques that can be used to perform a DoS attack. This attack tree is not inclusive, as it does not cover all possible ways to achieve the loss of availability.

In our assessment, we focus on vulnerabilities that exist in the system itself, especially the network stack. In what follows, we discuss various attacks that target the client, the server or the access point and present the associated experimental results. We want to emphasize that those attacks are not unique to the robotic system. A group of those attacks can target any device that is connected through a network (e.g. SYN flood). On the other hand, another group of those attacks can target devices that are connected through WiFi (e.g. de-authentication attack).

4.4.1 Application Layer Attack

This attack can be implemented on a wireless or wired network. It targets certain application on the server. The attacker sends a large number of requests to the target application on the server, overwhelming its processing and network resources. Typically, HTTP requests are used to exhaust the server. However, the robot does not run a web server, it runs an ARNL server on port 7272 and waits for connections from the client (MobileEyes). An attacker can exploit the ARNL server by initiating a large number of connections at the application level.

Experimental results: we performed a Distributed DoS (DDoS) attack, where seven adversary machines sent application level requests to the server. Initially the robot completed the ten-goal tour. However, after 15 minutes (on average) of the attack, the robot server freezes, old connections are lost and no new connections could be established.

4.4.2 TCP SYN Flood

This attack can be implemented on a wired or wireless network. A TCP SYN flood attack usually targets the operating system’s network handling capability of any server that is listening on a certain port. The

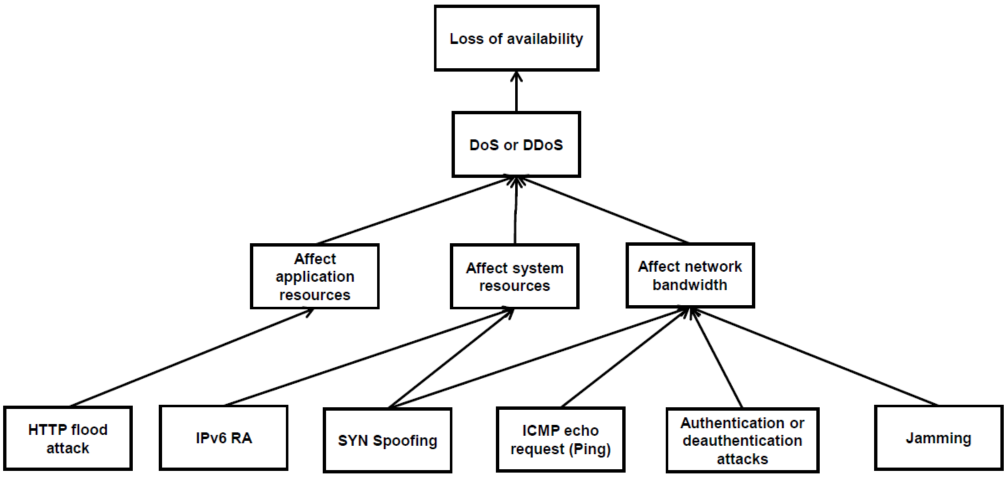


Figure 5. Attack tree with the loss of availability as the ultimate goal of the attacker.

attacker attempts to initiate a large number of TCP connections with the server, where each initiated TCP connection requires what is known as the three-way handshake process (SYN, SYN-ACK, ACK). The server allocates system resources to handle those connections. By overwhelming the server with the attacker's connections, the server will not have enough resources to handle legitimate connections with the client machine. Hence, the connection with the server can be lost, resulting in the loss of availability of the robot. However, this attack can be easily mitigated by blocking the attacker's IP address.

On the other hand, the attacker can use IP source address spoofing to improve the attack. In this case, the attacker uses spoofed IP addresses, so that the server responses are sent to devices that are unreachable (i.e., did not initiate the connection). This way, blocking the attacker's IP does not mitigate the attack. In summary, this kind of attack can result in overwhelming both the operating system of the robot and the robot network.

Experimental results: we performed this attack by sending spoofed TCP SYN flood targeting the robot on port 7272 (ARNL server). Whenever this attack is active, legitimate new connections could not be established from the client to the robot, so no commands could have been issued. Therefore, we could not evaluate the location of the robot as the command could not be issued in the first place. If the connection had already been established before the attack, then the attack fails.

4.4.3 IPv6 Router Advertisement Floods

In IPv6 Router Advertisement (RA) floods, the adversary floods the robot network with fake IPv6 RA packets or messages. Any computer, including the robot, running a vulnerable operating system (on which IPv6 routing is enabled), will be overwhelmed with those fake IPv6 RA packets. In order to perform this attack, the adversary should be connected to the same network of the victims (i.e., the robot network).

Experimental results: the results of this attack varied based on the vulnerabilities of the underlining operating system of the robot. We performed an IPv6 attack while running Windows 7 on the robot. Windows 7 is vulnerable to this attack. First, we issue a "tour goals" command from the client machine and immediately start the attack. Subsequently, after starting the attack, the robot operating system freezes, the robot stops before reaching goal 2 and the client loses connection with the robot. To restart the robot, it requires hard reset. Ubuntu operating system is not vulnerable to this type of attack. If client machine is also running a vulnerable operating system, then it will also freeze.

4.4.4 ICMP Echo Request

This attack works in the network layer. The attacker floods the robot or the client with ICMP echo request (ping) packets overwhelming the network of the robot. This degrades the network performance and causes legitimate traffic to be lost rendering the robot unavailable. To improve the attack, the attacker can use IP source address spoofing.

Experimental results: we performed a DDoS attack, where seven adversary machines sent spoofed ICMP echo requests to the server. Initially, the robot completed the-ten goal tour. However, after 45 minutes (on average) of the attack, the robot server freezes, old connections are lost and no new connections could be established.

4.4.5 De-authentication Attack

In a de-authentication attack, the attacker sends a de-authentication frame to the victim machine with a spoofed source address of the access point to terminate the victim's connection with the access point. This results in losing the WiFi connection and disconnecting the client from the server. This attack does not require encryption, so it can be performed by an outsider who can sniff the WiFi connection to get the MAC address of the victims and the access point [38].

Experimental results: we performed this attack after issuing a "tour goals" command from the client. The robot and client were immediately disconnected from the network. The client's map showed that the robot did not reach goal 2. However, this attack did not stop the robot from physically touring all the goals.

4.4.6 Jamming Attack

If the robot and client are connected using WiFi (IEEE 802.11), then they are susceptible to jamming attacks. This kind of attack requires that the attacker be in close proximity to the network under attack. The jammer has to operate in the 2.4 GHz or 5 GHz radio frequency which is used by the WiFi technology. This attack was not performed in our research, as it is a pure physical layer test which is beyond the scope of this paper.

4.4.7 Attack Summary

We specifically chose the previous attacks, because they demonstrate various ways to cause loss of availability (DoS). With the exception of the jamming attack, no special equipment is necessary and script kiddies can implement those attacks (e.g. using Kali Linux [39]). Table 1 summarizes attack characteristics and results. It demonstrates the physical domain of the attack (wired/wireless) and the targeted resources by the attack. Furthermore, Table 1 lists the impact of the attacks on the robot.

4.5 Semi-quantitative Risk Assessment Determination

As mentioned earlier in this section, it is convenient to use a semi-quantitative risk assessment approach. Table 2 represents the mapping between the qualitative and semi-quantitative values [15]. Following [15], we choose to use the semi-quantitative values in the range between 0 and 10, where 10 represents the maximum severity of vulnerability or impact.

The overall likelihood of any threat is assigned a value between 0 and 1 based on the likelihood of attack initiation and the likelihood of success of the initiated attack. Consequently, the risk level is the product of the overall likelihood and the impact of the attack, as demonstrated in Equation 2.

$$Risk = Overall\,Likelihood \times Impact; \tag{2}$$

where the overall likelihood of a threat is computed as a function of the likelihood of attack initiation and the likelihood of success of the initiated attack.

Now, we discuss how to generate the risk table, which summarizes the risk assessment process. We used the adversarial risk assessment template proposed in [15]. Table 3 presents the end result of the risk assessment process for the threat events under study. Following, each item in Table 3 is discussed (for more details about those items please refer to tables D-3, D-4, D-5, E-4, F-2, F-5 and H-3 in [15]):

Threat Event: refers to the threat that is currently being analyzed.

1) *Threat Sources*: refer to the threat source, which can be an insider, outsider, trusted insider or privileged insider [15]. Some attacks require the attacker to communicate with the robot, client or access point. This implies that the attacker has to be on the same network of those devices (unless the robot is configured to be remotely accessed, like having a public IP address). This applies to application level attacks, TCP SYN floods, IPv6 RA floods and ICMP echo requests.

Table 1. Summary of threat characteristics and experimental results

Threat	**Wireless or wired**	**Target resources**	**Impact on the robot**
Application level attack	Wireless	Application resources	Connection with the robot is lost. No new connections could be established.
TCP SYN flood attack	Both	System resources and/or network bandwidth	No new legitimate client connections could be established with the robot.
IPv6 RA attack	Both	System resources	OS (Widows 7) freezes, robot stops before goal 2 and robot disconnected from client.
ICMP echo request flood	Both	Network bandwidth	Connection with the robot is lost. No new connections could be established.
De-authentication attack	Wireless	System resources	Robot disconnected from client before reaching goal 2 and robot is moving without supervision.
Jamming attack	Wireless	Network bandwidth	No connection between the client and the robot.

Table 2. Mapping of the qualitative levels to the semi-quantitative levels

Qualitative level	Very High	High	Moderate	Low	Very Low
Semi-quantitative level	10	8	5	2	0

On the other hand, de-authentication and jamming attacks can be conducted by outsiders with close proximity to the target machines.

2) *Capability*: this is one of the characteristics of the threat source. An attacker with high capability is one with high level of expertise and is well-resourced. Because of the increasing interest in cyber-security, there is an increasing number of highly capable threat sources, especially in a scientific and research environment. The reader should note that the attacks we are implementing can be implemented by a person with moderate or low capabilities.

3) *Intent*: this is one of the characteristics of the threat source. The adversary seeks to undermine critical functions of the system and may result in physical damage by causing loss of availability.

4) *Targeting*: this is one of the characteristics of the threat source. The adversary targets a specific mission or function within an organization (i.e., the target of the attack is not random; however, the threats discussed in this work are not unique to the robots).

5) *Relevance*: indicates how relevant the threat event is to the system under study. For example, if the threat event already happened in the system, then it is confirmed. As we do not have evidence that those threats happened (i.e., reported in the literature as real attack) on a PeopleBot robot, we rank them as possible.

6) *Likelihood of attack initiation*: refers to the likelihood that the adversary will initiate the attack. Hence, it depends on adversary intend, capability and targeting [15]. To find this likelihood, we compute the normalized average of those three values as shown in Equation 3.

$$normalized\ average = \frac{computed\ average}{10} \tag{3}$$

7) *Severity of vulnerabilities*: all of our implemented attacks make use of the vulnerabilities in the communication link with the robot, which are exploitable and exposed. Application level, TCP SYN floods and ICMP echo request can be mitigated by blocking the source IP address of the attacker. However, if IP address spoofing is used, then protecting against those attacks will be difficult. As such, the severity of the attacks is ranked high. IPv6 RA floods can be easily mitigated by disabling IPv6 routing, so it is ranked as moderate. In addition, Windows operating

systems are highly affected by this attack, while Linux operating systems are moderately affected, because they only take the first 15 route advertisements and ignore the remaining. De-authentication and jamming attacks are very difficult to be stopped [40], [41].

8) *Pervasiveness of predisposing conditions*: vulnerabilities that apply to all robots running the same setup are ranked high.

9) *Likelihood Initiated Attack Succeeds*: This item depends on the threat source capability, severity of the vulnerability and predisposing conditions. To compute the probability of a successful initiated attack, we average those three factors and then normalize them.

10) *Overall Likelihood*: this is a combination of the likelihood of attack initiation and the likelihood of the successful initiated attack. In order to capture the effect of both likelihoods, we average the two values.

11) *Level of impact on the robot*: severe or catastrophic effect on the robot system means high impact. All the attacks discussed in this paper cause high impact, which is the loss of availability of the robot.

12) *Risk*: this is the final risk assessment measure, which is the product of the overall likelihood and the level of impact (Equation 2). The risk value represents an estimation of the risk of each threat and can be used to prioritize risk handling; for example, (application level attack):

- The threat source characteristics are all ranked high and hence they were assigned a value of 8 based on Table 2.
- The likelihood of attack initiation is computed based on Equation 3 using the threat source characteristics values (i.e., $\frac{\frac{8+8+8}{3}}{10} = 0.8$).
- The severity of vulnerabilities and the pervasiveness of predisposing conditions were all ranked high and hence they are assigned a value of 8.
- The likelihood that the initiated attack succeeds is computed based on Equation 3 using the threat source capability, severity of vulnerabilities and the pervasiveness of predisposing conditions values (i.e., $\frac{\frac{8+8+8}{3}}{10} = 0.8$).
- The overall likelihood is computed by taking the average of the likelihood of attack initiations and the likelihood that the initiated attack succeeds (i.e.,$\frac{0.8+0.8}{2} = 0.8$).
- The level of impact on the robot is ranked high and thus is assigned a value of 8.
- The risk level is computed based on Equation 2 (i.e., $0.8 \times 8 = 6.4$).

In the following section, we discuss the physical consequences for the loss of availability on the robot.

5. Discussion

In this section, we discuss the physical consequences of the loss of availability on the robot. Since this is an application-specific process, we consider several applications of the robot. We consider the loss of availability as the impact of the attack in the risk assessment process. The impact values in the risk assessment process can be assigned based on the specific robotic application. For example, if the robot freezes in a high traffic area (e.g. airport application), then this may cause more damage than if it continues with the current task. On the other hand, it may be safer for the robot to stop and abort its current task if moving with precise supervision is required (e.g. industrial applications).

5.1 Hospital Application

This application is for the case when the robot is operated in a hospital that is fully covered by a wireless LAN. In this application, the robot is assumed to be used in two critical missions: creating a 2D map of the hospital and performing time-sensitive tasks (e.g. delivering medical supplies and equipment) [42]. Both missions require the robot to be remotely controlled and monitored by a human operator through the wireless LAN. Creating a map requires logging lots of data generated from the 2D laser range finder sensor and the wheel encoders of the robot [33]. The operator connects to the robot through an SSH

Table 3. Risk Assessment Table of the PeopleBot robot

1	2	3	4	5	6	7	8	9	10	11	12	13
Threat Event	**Threat Sources**	**Threat Source Characteristics**			**Relevance**	**Likelihood of attack initiation**	**Severity of vulnerabilities**	**Pervasiveness of predisposing conditions**	**Likelihood of Initiated Attack Success**	**Overall Likelihood**	**Level of impact on robot**	**Risk**
		Capability	**Intent**	**Targeting**								
Application level attack	Insider	8	8	8	Possible	0.80	8	8	0.80	0.80	8	6.40
TCP SYN flood attack	Insider	8	8	8	Possible	0.80	8	8	0.80	0.80	8	6.40
IPv6 RA floods	Insider	8	8	8	Possible	0.80	2	8	0.60	0.70	8	5.60
ICMP echo request attack	Insider	8	8	8	Possible	0.80	8	8	0.80	0.80	8	6.40
De-authentication attack	Insider or outsider	8	8	8	Possible	0.80	10	8	0.87	0.84	8	6.72
Jamming attack	Insider or outsider	8	8	8	Possible	0.80	10	8	0.87	0.84	8	6.72

session and runs a data logging program, such as “sickLogger”[43]. Moreover, the operator runs a monitoring program to control the robot movement using image sequences from the camera sensor. While running these applications, multiple physical consequences are possible if the robot becomes unavailable as a result of exploiting the attacks discussed in Subsection 4.4. First, the robot may get physically damaged due to open stairs in the hospital environment. Second, sudden accidents may cause human injuries or damage critical equipment.

On the other hand, performing time-sensitive delivery tasks requires the operator and the robot to connect using client-server programs; “MobileEyes” and “arnlServer”[43]. The robot runs the ARNLServer program which loads the hospital map in the robot’s memory then waits for commands from the client program. The client runs the MobileEyes program, which connects to the ARNLServer and navigates the robot to destinations within the map using IR, bumpers, camera, sonars and laser range finder sensors. One critical physical consequence of loss of availability in this scenario is delaying or preventing the robot delivering urgent medical supplies to the hospital operation room. Another critical consequence is that the robot may get physically hijacked.

5.2 Airport Application

This application considers the case when the robot is operated at airports to perform two critical tasks: delivering passengers’ luggage to their respective terminals [44] and carrying out security checks for travelers [45]. To achieve the first task, the robot needs to create an airport map for luggage delivery. Creating the airport map can be done exactly the same fashion in the hospital scenario; hence, similar physical consequences might occur after a successful attack that leads to loss of availability. Once the map is constructed, the “MobileEyes” and “ARNLServer” client-server applications are used to load the map in the robot’s memory and send navigation commands to the robot for delivery purposes. Losing availability while delivering passengers’ luggage might result in luggage being delayed, which causes customer dissatisfaction and impacts the airline public image. More severely, the luggage might maliciously get stolen, replaced or delivered to incorrect terminals.

To achieve the second task (i.e., performing security checks for travelers), the robot uses its high-definition cameras and facial recognition software probably running on the operator's computer. Losing availability while carrying out such critical task has the dire consequence of allowing suspects to escape the security check.

5.3 Industrial Application

This application considers the case when the robot is operated at an industrial warehouse for helping in the production of expensive instruments and equipment. This is one of the critical scenarios, as the robot needs to make precise movements and actions. Losing availability of the robot could result in a financial impact. In this case, financial losses can be huge due to production halt, business disruption and replacement or remediation costs. It is clear that the physical consequences described above can lead to catastrophic results. Security experts need to carefully consider taking well-studied countermeasures to reduce or even prevent attacks. Surely this needs to be considered per each application of the robotic platform.

5.4 Mitigating Robot Cyber-Security

In this subsection, we suggest some recommendations as possible mitigation techniques for the loss of availability of the robot. It is important to mention that building a secure robot is not a simple task. However, considering and implementing the following recommendations can highly improve the cyber-security of the robot.

- Encrypt the robot communications.
- Assure that only authorized users have access to the robot network, on-board computer, services and functionality.
- Install operating systems updates to fix known vulnerabilities.
- Invest in cyber-security education for everyone using the robot.
- Enforce a backup plan policy in case the robot becomes unavailable.
- Enable SYN Cookies to protect against TCP SYN floods [46].

5.5 Limitations

The main limitation of this approach is the dependency on identifying the vulnerabilities of the system. This requires deep knowledge of all the systems integrated to operate the robot, including hardware, software, operating systems, communications, …etc.

6. CONCLUSIONS

In this paper, we identified, estimated and prioritized the risks associated with attacks targeting the availability of the robotic system.

The paper discussed several attacks that can result in losing the availability of the PeobleBot robot. We setup a ten-goal test area in our research lab. The robot is commanded to tour the ten goals while being attacked. We discussed the experimental results from each attack. The paper presented an impact-oriented analysis approach to assess the risk of these attacks as demonstrated in Table 3. We discussed the physical impacts for losing the availability of the robot while performing several critical applications. The severity of the physical impacts raises the flag of the requirement of considering effective cyber-security countermeasures.

Future work could focus on extending the semi-quantitative risk assessment to a quantitative assessment. In addition, one area of research is to examine attacks on integrity and confidentiality of robotic systems. We also plan to analyze the resilience of robotic systems to cyber-physical attacks. By analyzing the resilience, we investigate how the robotic system responds and recovers from failures that are caused by cyber-physical attacks.

ACKNOWLEDGEMENTS

This research was supported by a grant from the Deanship of Scientific Research at the Hashemite University, Zarqa, Jordan.

REFERENCES

[1] I. Priyadarshini, "Cyber Security Risks in Robotics," Detecting and Mitigating Robotic Cyber Security Risks, IGI Global, pp. 333–348, 2017.

[2] J. L. Jones, N. E. Mack, D. M. Nugent and P. E. Sandin, Autonomous Floor-cleaning Robot, 2009.

[3] B. Hannaford et al., "Raven-II: An Open Platform for Surgical Robotics Research," IEEE Transactions on Biomedical Engineering, vol. 60, no. 4, pp. 954–959, 2013.

[4] H. Alemzadeh, D. Chen, X. Li, T. Kesavadas, Z. T. Kalbarczyk and R. K. Iyer, "Targeted Attacks on Teleoperated Surgical Robots: Dynamic Model-based Detection and Mitigation," Proc. of the 46th Annual IEEE/IFIP Inter. Conf. on Dependable Systems and Networks (DSN), pp. 395–406, 2016.

[5] C. Cerrudo and L. Apa, "Hacking Robots before Skynet1," IOActive Website, 2017.

[6] A. Khalid, P. Kirisci, Z. H. Khan, Z. Ghrairi, K.-D. Thoben and J. Pannek, "Security Framework for Industrial Collaborative Robotic Cyber-physical Systems," Computers in Industry, vol. 97, pp. 132–145, 2018.

[7] A. Y. Javaid, W. Sun, V. K. Devabhaktuni and M. Alam, "Cyber Security Threat Analysis and Modeling of an Unmanned Aerial Vehicle System,", Proc. of the IEEE Conference on Technologies for Homeland Security (HST), pp. 585–590, 2012.

[8] A. J. Kornecki and Z. Janusz, "Threat Modeling for Aviation Computer Security," CrossTalk, vol. 21, 2015.

[9] A. Sanfeliu Cortés, "URUS: Ubiquitous Networking Robotics for Urban Settings," Cognitive Systems Industry Day (CSID), 2008.

[10] T. Jason, S. C. Chan, G. Ngai, J. C. Cheung and V. T. Ng, "Dynamic Collaborative Robotic Platform-A Brief Introduction," Proc. of the 13th International Conference on Computer Supported Cooperative Work in Design (CSCWD 2009), pp. 125–130, 2009.

[11] Y.-H. Wei, Q. Leng, S. Han, A. K. Mok, W. Zhang and M. Tomizuka, "RT-WiFi: Real-time High-speed Communication Protocol for Wireless Cyber-physical Control Applications," Proc. of the 34th IEEE on Real-Time Systems Symposium (RTSS), pp. 140–149, 2013.

[12] D. Quarta, M. Pogliani, M. Polino, F. Maggi, A. M. Zanchettin and S. Zanero, "An Experimental Security Analysis of an Industrial Robot Controller," IEEE Symposium on Security and Privacy (SP), pp. 268–286, 2017.

[13] "PeopleBot," [Online], Available: http://www.mobilerobots.com/ResearchRobots/PeopleBot.aspx.

[14] K. Ahmad Yousef, A. AlMajali, R. Hasan, W. Dweik and B. Mohd, "Security Risk Assessment of the PeopleBot Mobile Robot Research Platform," Proc. of the International Conference on Electrical and Computing Technologies and Applications (ICECTA), pp. 1–5, 2017.

[15] R. Blank and P. Gallagher, "NIST Special Publication 800-30 Revision 1 Guide for Conducting Risk Assessments," National Institute of Standards and Technology, 2012.

[16] J. Holliman, M. Zhivich, R. Khazan, A. Swiston and B. Telfer, "Building Low-power Trustworthy Systems: Cyber-security Considerations for Real-time Physiological Status Monitoring," Proc. of the IEEE Military Communications Conference (MILCOM 2016), pp. 1083–1089, 2016.

[17] I. Kateeb and M. Almadallah, "Risk Management Framework in Cloud Computing Security in Business and Organizations," IAJC/ISAM Joint International Conference, 2014.

[18] E. Moradian and M. Kalinina, "Decision Support for Assessment of IT-Security Risks," Proceedings of the International Conference on Security and Management (SAM), p. 1, 2013.

[19] T. Bonaci, J. Herron, T. Yusuf, J. Yan, T. Kohno and H. J. Chizeck, "To Make a Robot Secure: An Experimental Analysis of Cyber Security Threats Against Teleoperated Surgical Robots," arXiv preprint arXiv:1504.04339, 2015.

[20] G. Vasconcelos, G. Carrijo, R. Miani, J. Souza and V. Guizilini, "The Impact of DoS Attacks on the AR.Drone 2.0," 2016 XIII Latin American Robotics Symposium and IV Brazilian Robotics Symposium (LARS/SBR), pp. 127–132, 2016.

[21] N. Bezzo, J. Weimer, M. Pajic, O. Sokolsky, G. J. Pappas and I. Lee, "Attack Resilient State Estimation for Autonomous Robotic Systems," Proc. of the IEEE/RSJ Inter. Conf. on Intelligent Robots and

Systems, pp. 3692–3698, 2014.

[22] L. T. Batson, D. R. Wimmer Jr et al., Unmanned Tactical Autonomous Control and Collaboration Threat and Vulnerability Assessment, PhD Thesis, Monterey, California: Naval Postgraduate School, 2015.

[23] A. Jones and J. Straub, "Using deep learning to detect network intrusions and malware in autonomous robots," SPIE Defense+ Security, pp. 1018505–1018505, 2017.

[24] F. Maggi, D. Quarta, M. Pogliani, M. Polino, A. M. Zanchettin and S. Zanero, "Rogue Robots: Testing the Limits of an Industrial Robot's Security," Technical Report, Trend Micro, Politecnico di Milano, 2017.

[25] F. J. R. Lera, C. F. Llamas, Á. M. Guerrero and V. M. Olivera, "Cybersecurity of Robotics and Autonomous Systems: Privacy and Safety," Robotics-Legal, Ethical and Socioeconomic Impacts, InTech, 2017.

[26] T. Vuong et al., Cyber-physical Intrusion Detection for Robotic Vehicles, PhD Thesis, University of Greenwich, 2017.

[27] G. Loukas, T. Vuong, R. Heartfield, G. Sakellari, Y. Yoon and D. Gan, "Cloud-based Cyber-Physical Intrusion Detection for Vehicles Using Deep Learning," IEEE Access, vol. 6, pp. 3491–3508, 2018.

[28] Q. Chen, R. K. Abercrombie and F. T. Sheldon, "Risk Assessment for Industrial Control Systems Quantifying Availability Using Mean Failure Cost (MFC)," Journal of Artificial Intelligence and Soft Computing Research, vol. 5, no. 3, pp. 205–220, 2015.

[29] D. Dominic, S. Chhawri, R. M. Eustice, D. Ma and A. Weimerskirch, "Risk Assessment for Cooperative Automated Driving," Proceedings of the 2nd ACM Workshop on Cyber-Physical Systems Security and Privacy, pp. 47–58, 2016.

[30] K. M. Ahmad Yousef, A. AlMajali, S. A. Ghalyon, W. Dweik and B. J. Mohd, "Analyzing Cyber-Physical Threats on Robotic Platforms," Sensors, vol. 18, no. 5, p. 1643, 2018.

[31] H. Hüttenrauch, K. S. Eklundh, A. Green and E. A. Topp, "Investigating Spatial Relationships in Human-robot Interaction," Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 5052–5059, 2006.

[32] A. Chella et al., "A BCI Teleoperated Museum Robotic Guide," Proc. of the International Conference on Complex, Intelligent and Software Intensive Systems (CISIS'09), pp. 783–788, 2009.

[33] H. Kwon, K. M. A. Yousef and A. C. Kak, "Building 3D Visual Maps of Interior Space with a New Hierarchical Sensor Fusion Architecture," Robotics and Autonomous Systems, vol. 61, no. 8, pp. 749–767, 2013.

[34] K. M. Ahmad Yousef, J. Park and A. C. Kak, "Place Recognition and Self-localization in Interior Hallways by Indoor Mobile Robots: A Signature-based Cascaded Filtering Framework," Proc. of the IEEE/RSJ Inter. Conf. on Intelligent Robots and Systems (IROS 2014), pp. 4989–4996, 2014.

[35] I.-H. Kuo, E. Broadbent and B. MacDonald, "Designing a Robotic Assistant for Healthcare Applications," Proc. of the 7th Conference of Health Informatics, New Zealand, Rotorua, 2008.

[36] G. D. Morais, L. C. Neves, A. A. Masiero and M. C. F. de Castro, "Application of Myo Armband System to Control a Robot Interface, " Biosignals, pp. 227–231, 2016.

[37] K. M. Ahmad Yousef, B. J. Mohd, K. Al-Widyan and T. Hayajneh, "Extrinsic Calibration of Camera and 2D Laser Sensors without Overlap," Sensors, vol. 17, no. 10, p. 2346, 2017.

[38] "Deauthentication Attack," [Online], Available: https://www.aircrack-ng.org/doku.php?id= deauthentication.

[39] "Kali Linux," [Online], Available: https://www.kali.org/.

[40] M. J. Handley and E. Rescorla, "Internet Denial-of-service Considerations," 2006.

[41] V. Dey, V. Pudi, A. Chattopadhyay and Y. Elovici, "Security Vulnerabilities of Unmanned Aerial

Vehicles and Countermeasures: An Experimental Study," Proc. of the 17th Inter. Conf. on Embedded Systems (VLSID) and the 31st Inter. Conf. on VLSI Design, pp. 398–403, 2018.

[42] A. G. Ozkil, Z. Fan, S. Dawids, H. Aanes, J. K. Kristensen and K. H. Christensen, "Service Robots for Hospitals: A Case Study of Transportation Tasks in a Hospital," Proc. of the IEEE International Conference on Automation and Logistics (ICAL'09), pp. 289–294, 2009.

[43] "Creating A Laser Map for ARNL," [Online], Available: http://robots.mobilerobots.com/wiki/Creating_A_Laser_Map_for_ARNL.

[44] "This is Real Life: Robotics Company Cyberdyne Introducing ‘Service’ Robots with Artificial Intelligence," [Online], Available: http://nationalpost.com/news/world/this-is-real-life-robotics-company-cyberdyne-introducing-service-robots-with-artificial-intelligence.

[45] "Rise of the Airport Robots," [Online], Available: https://www.aerosociety.com/news/rise-of-the-airport-robots/.

[46] D. J. Bernstein, "SYN cookies," [Online], Available: http://cr.yp.to/syncookies.html.

ملخص البحث:

أصبحت الروبوتات تتكامل بشكل متزايد في حياتنا اليومية، حيث تقدم الخدمات في التطبيقات المدنية والصناعية والعسكرية. العديد من هذه التطبيقات تتطلب أن يتم تشغيل الروبوتات والتحكم فيها عن بعد من خلال قنوات الاتصال. وهذا يجعل نظام الروبوت عرضة لطبقة من الهجمات التي تستهدف الاتصال بين العميل المسيطر والروبوت، مما يجعل الروبوت غير متوفر. الهدف من هذه الدراسة هو تحديد وتقييم المخاطر المرتبطة بالهجمات التي تستهدف توفر الروبوت. من أجل تحقيق هدفنا، قمنا بإجراء تقييم مخاطر شبه كمي موجه نحو تأثير فقدان توفر الروبوت (PeopleBot™). قمنا بإجراء مجموعة من التجارب التي استخدمنا فيها العديد من الهجمات المعروفة التي يمكن أن تستهدف وتؤثر على توافر الروبوت. وقد أثبتت النتائج أن الهجمات يمكن أن تؤدي إلى فقدان توفر الروبوت، الأمر الذي يؤدي بدوره إلى عواقب خطيرة.

JJCIT Annual List of Reviewers (2018)

Name, *Affiliation*, Country

Abdelhak Lakhouaja, *Mohammed First University*, Morocco

Abdellah Yousfi, *Mohamed V University*, Morocco

Abdul Malik Al-Salman, *King Saud University*, Saudi Arabia

Abdul Rahim Ahmad, *UNITEN*, Malaysia

Ahmad T. Al-Taani, *Yarmouk University*, Jordan

Ahmed Rafea, *AUC*, Egypt

Alessandro Rizzi, *University of Milan*, Italy

Ali Younis Maqousi, *University of Petra*, Jordan

Anqi Cui, *RSVP Technologies Inc.*, Canada

Antonio Navarro, *Aveiro University*, Portugal

Artur Babiarz, *Silesian University of Technology*, Poland

Benjamin Aziz, *University of Portsmouth*, UK

Cheng Li, *William & Mary*, USA

Chunmei Liu, *Howard University*, USA

Danilo Mandi, *Imperial College London*, UK

Degan Zhang, *TU TE*, China

Delina Beh Mei, *UniKL*, Malaysia

Devasis Pardhan, *Acharya Institutes*, India

Driss Namly, *Mohammed V University*, Morocco

Duangjai Jitkongchuen, *Dhurakij Pundit* University, Thailand

Duygu Sinanc Terzi, *Gazi University*, Turkey

E. Uzun, *TOBB ETÜ*, Turkey

El Houssein Chouaib Harik, *NIBIO*, Norway

Emad M. Al-Shawakfa, *Yarmouk University*, Jordan

Evandro Cunha, *UFMG*, Brazil

Fadoua Khennou, *Sidi Mohamed Ben Abdellah University*, Morocco

Mehmet Fatih ÇAĞLAR, *Süleyman Demirel University*, Turkey

Fawaz S.Al-Anzi, *Kuwait University*, Kuwait

Firas A. Jassim, *Harvard University*, USA

Floriano De Rango, *University of Calabria*, Italy

Frans Hendrik Botes, *CPUT*, South Africa

George Pallis, *University of Cyprus*, Cyprus

Ghassan G. Kanaan, *Amman Arab University*, Jordan

Gudivada Viswanadh Raviteja, *GITAM University*, India

Gunasekaran Manogaran, *University of California*, USA

Hamid Reza Hassani, *Shahed University*, Iran

Hani Abu-Salem, *University of South Carolina*, USA

Heekuck Oh, *Hanyang University*, Korea

Hosein M. Golshan Mojdehi, *University of Denver*, USA

Hossam M. Zawbaa, *Beni-Suef University*, Egypt

Ibéria Medeiros, *University of Lisboa*, Portugal

Imad Zeroual, *Mohamed First University*, Morocco

Inès Zribi, *University of Sfax*, Tunisia

Ioannis Agrafiotis, *University of Oxford*, UK

Irini Fundulaki, *Foundation for Research and Technology*, Greece

Iztok Jr Fister, *University of Maribor*, Slovenia

Jamal Nahar Bani Salamehj, *Mutah University*, Jordan

Jawad K. Ali, *University of Technology*, Iraq

Jin-Fu Li, *National Central University*, Taiwan

Jorge Dias, *Khalifa University*, UAE

K. Anitha Kumari, *PSG College of Technology*, India

Kalman Graffi, *HHUD*, Germany

Ke Gu, *BJUT*, China

Lilly Suriani Affendey, *UPM*, Malaysia

Luanna Lopes Lobato, *Federal University of Goias*, Brazil

M. B. R. Murthy, *RGUKT*, India

Maha Alrabiah, *Al Imam Mohammad ibn Saud Islamic University*, Saudi Arabia

Mahmoud Al-Ayyoub, *JUST*, Jordan

Mahmoud El-Haj, *Lancaster University*, UK

Majed Said AbuSafiya, *AAU*, Jordan

Manisha Malhotra, *Chandigarh University*, India

Massimo Tosa, *Technical University of Kaiserslautern*, Germany

Meenupriya Swaminathan, *Northeastern University*, USA

Mehmet A. Belen, *Yıldız Technical University*, Turkey

Michel R. V. Chaudron, *University of Gothenburg*, Sweden

Minxian Xu, *University of Melbourne*, Australia

Mohammad Tariqul Islam, *UKM*, Malaysia

Mohamed Mahmoud, *HICIT*, *Shorouk Academy*, Egypt

Mohammad H. Alomari, *ASU*, Jordan

Mohammed Naji Al-Kabi, *Zarqa University*, Jordan

Mousa I. Hussein, *UAE University*, UAE

Nada Ghneim, *Syrian Virtual University*, Syria

Najmiah Radiah Mohamad, *UTEM*, Malaysia

Narinder Singh, *Punjabi University*, India

Nizar Habash, *New York University*, USA

Osama M.F. Abu-Sharkh, *PSUT*, Jordan

Osama Mohamed Elrajubi, *Misurata University*, Libya

P. S. Ashtankar, *KITS*, India

Paolo Rosso, *Technical University of Valencia*, Spain

Pei Luo, *Northeastern University*, USA

Peipei Wang, *NC SU*, USA

Petra Leimich, *Edinburgh Napier University*, UK

Prayoot Akkaraekthalin, *KMUTNB*, Thailand

Priti Sehgal, *University of Delhi*, India

Qammer H. Abbasi, *University of Glasgow*, UK

Qi Han, *HIT*, China

Qi Jiang, *Xidian University*, China

Quan Leng, *University of Texas*, USA

Rabii Ayed, *ISGS*, Tunisia

Radu-Emil Precup, *Polytechnic of Timisoara*, Romania

Razali Ngah, *UTM*, Malaysia

Riad Sonbol, *HIAST*, Syria

Richard McClatchey, *University of the West of England*, UK

Rodrigo Roman, *University of Málaga*, Spain

Rohit Salgotra, *TIET*, India

Roshan Ragel, *University of Peradeniya*, Sri Lanka

S. K. Bharti, *University of Delhi*, India

Saad Mustafa, *COMSATS Institute of Information Technology*, Pakistan

Sajad Mohammad-Ali-Nezhad, *University of Qom*, Iran

Salvatore Sessa, *University of Naples Federico II*, Italy

Sandeep Kumar Samal, *Georgia Institute of Technology*, USA

Sankalap Arora, *DAV University*, India

Sarawuth Chaimool, *KMUTNB*, Thailand

Saru Kumari, CCS *University*, India

Sayyed Arif Ali, *Mewar University*, India

Shahrul Azman, *UKM*, Malaysia

Shailesh D. Kamble, YCCE, India

Shubham Gupta, *Facebook*, USA

Simona Bernardi, *University of Zaragoza*, Spain

Syed Ahmed Raza, *Western University*, Canada

Thomas De Cnudde, *KU Leuven*, Belgium

Thomas Pasquier, *University of Cambridge*, UK

Tolga Ensari, *Istanbul University*, Turkey

Vladimir B. Kovacevic, *University of Belgrade*, Serbia

Weiqing Sun, *University of Toledo*, USA

Xavier Luciani, *University of Toulon*, France

Xuechao Ming, *TU TE*, China

Yaser Al-Lahham, *Zarqa University*, Jordan

Yaser Jararweh, *JUST*, Jordan

Yin Minn Pa Pa's, *Yokohama National* University, Japan

Yu Zhanga, *Stanford University*, USA

Yuntao Wu, *WIT*, China

Zeashan Hameed Khan, *Bahria University*, Pakistan

Zhaopeng Cui, *ETH Zurich*, Switzerland

Zhaoqiang Xia, *NPU*, China

Zhen Chen, *University of Rhode Island*, USA

Zijiang Hao, *William & Mary*, USA

Zoltán Adam Mann, *Paluno*, Germany

Princess Sumaya University for Technology
Since 1991 تأسست
PSUT
جامعة الأميرة سمية للتكنولوجيا

صندوق دعم البحث العلمي
Scientific Research Support Fund

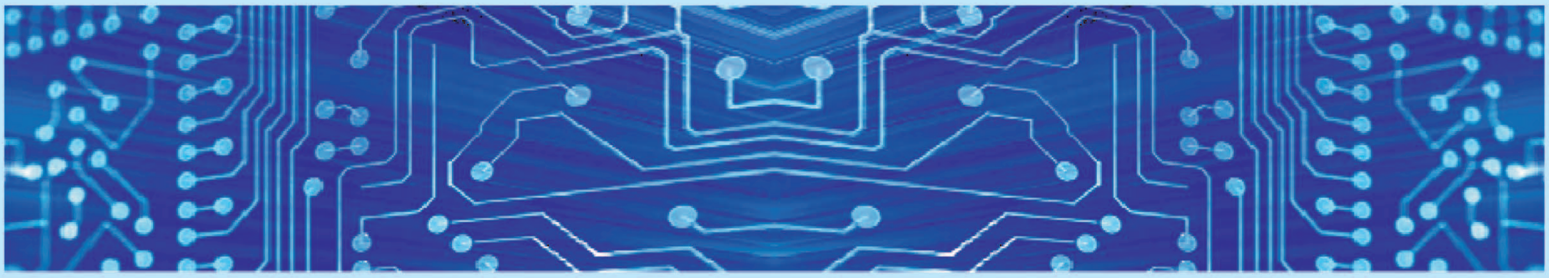
المجلة الأردنية للحاسوب و تكنولوجيا المعلومات

ISSN 2415 - 1076 (Online)
ISSN 2413 - 9351 (Print)

العدد ٣ المجلد ٤ كانون الأول ٢٠١٨

JJCIT

www.jjcit.org jjcit@psut.edu.jo



مجلة علمية عالمية متخصصة محكمة

تصدر بدعم من صندوق دعم البحث العلمي