




## Article

# AI-Based Resource Allocation Techniques in Wireless Sensor Internet of Things Networks in Energy Efficiency with Data Optimization

Quazi Warisha Ahmed <sup>1</sup>, Shruti Garg <sup>1</sup>, Amrita Rai <sup>2</sup>, Manikandan Ramachandran <sup>3</sup>, Noor Zaman Jhanjhi <sup>4,\*</sup>, Mehedi Masud <sup>5</sup> and Mohammed Baz <sup>6</sup>

<sup>1</sup> Computer Science & Engineering, Birla Institute of Technology Mesra, Ranchi 835215, India; wareeshahmed789@gmail.com (Q.W.A.); gshruti@bitmesra.ac.in (S.G.)

<sup>2</sup> Department of Electronics & Communication Engineering, GL Bajaj Institute of Technology and Management, Knowledge Park III, Greater Noida 201306, India; amritarai25@hotmail.com

<sup>3</sup> School of Computing, SASTRA Deemed University, Thanjavur 613401, India; srmanimt75@gmail.com

<sup>4</sup> School of Computer Science, SCS, Taylor's University, Subang Jaya 47500, Malaysia

<sup>5</sup> Department of Computer Science, College of Computers and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia; mmasud@tu.edu.sa

<sup>6</sup> Department of Computer Engineering, College of Computers and Information Technology, Taif University, P.O. Box. 11099, Taif 21994, Saudi Arabia; mo.baz@tu.edu.sa

\* Correspondence: noorzaman.jhanjhi@taylors.edu.my

**Abstract:** For the past few years, the IoT (Internet of Things)-based restricted WSN (Wireless sensor network) has sparked a lot of attention and progress in order to attain improved resource utilisation as well as service delivery. For data transfer between heterogeneous devices, IoT requires a stronger communication network and an ideally placed energy-efficient WSN. This study uses deep learning architectures to provide a unique resource allocation method for wireless sensor IoT networks with energy efficiency as well as data optimization. EE (Energy efficiency) and SE (spectral efficiency) are two competing optimization goals in this case. The network's energy efficiency has been improved because of a deep neural network based on whale optimization. The heuristic-based multi-objective firefly algorithm was used to optimise the data. This proposed method is applied to optimal power allocation and relay selection. The study is for a cooperative multi-hop network topology. The best resource allocation is achieved by reducing overall transmit power, and the best relay selection is accomplished by meeting Quality of Service (QoS) standards. As a result, an energy-efficient protocol has been created. The simulation results demonstrate the suggested model's competitive performance when compared to traditional models in terms of throughput of 96%, energy efficiency of 95%, QoS of 75%, spectrum efficiency of 85%, and network lifetime of 91 percent.

**Keywords:** wireless sensor network; Internet of Things; resource allocation; energy efficiency; data optimization; deep learning



**Citation:** Ahmed, Q.W.; Garg, S.; Rai, A.; Ramachandran, M.; Jhanjhi, N.Z.; Masud, M.; Baz, M. AI-Based Resource Allocation Techniques in Wireless Sensor Internet of Things Networks in Energy Efficiency with Data Optimization. *Electronics* **2022**, *11*, 2071. <https://doi.org/10.3390/electronics11132071>

Academic Editors: Seokjoo Shin and Haeyoung Lee

Received: 29 May 2022

Accepted: 23 June 2022

Published: 1 July 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

One of the most important study topics in the cloud is resource allocation, which aims at increasing service provider profitability and achieving customer satisfaction by meeting promised SLA conditions [1]. SLA (Service Level Agreements) must be signed by Service Providers and Cloud Users to assure Quality of Service (QoS) [2]. Resource allocation has been considered one of the most significant topics to address when dealing with SLA situations. Because the load on the physical server's changes over time, resource allocations must be managed dynamically. Dynamic resource allocation is exceedingly difficult, especially when QoS needs to change over time while considering processor availability and minimising processor idle time. A WSN is a network of several sensor nodes that are typically put in remote places to monitor environmental characteristics. Sensors such as

acoustic, pressure, motion, image, chemical, weather, pressure, temperature, and optical sensors are installed in the sensor nodes (SN). WSNs have a wide range of applications due to the diversity of SN, ranging from healthcare to military, defense, agricultural, and everyday life. Despite their large uses, WSNs have numerous common issues such as restricted energy sources, processing speed, memory, and communication bandwidth, causing SN performance to degrade as well as network lifetime to decrease [3]. Creating distinct algorithms for various purposes is a difficult endeavor. WSN designers must pay special attention to concerns such as data aggregation, clustering, routing, localization, fault detection, task scheduling, and event tracking, among others.

Wireless sensor nodes are small devices that detect atmospheric conditions including pressure, temperature, and humidity. They have a memory device to store the data and a channel to transfer it to BSAs well as other devices. They are frequently dispersed, depending on the number of nodes used to collect data. Many studies have been previously conducted [4] that address these problems by applying methodologies derived from signal communication theory in telephony, with the primary goal of ensuring reliable data delivery without noise. Because there is no way to provide continuous power to sensors using a battery as a power source, researchers must focus on energy efficiency. Because of the limited energy sources, the sensor node has a short lifespan, which reduces the system's network lifetime. Machine learning (ML) methods are known for their self-experiencing nature as well as the fact that they do not require reprogramming [5]. ML is a useful approach that allows for efficient, dependable, and cost-effective computing. Supervised learning, unsupervised learning, and RL are the three main forms. It has been discovered that machine learning technologies are effective in resolving key WSN difficulties. In the realms of IoT, M2M and CPS, these approaches have proven to be beneficial. ML may learn from a generalized structure and propose a generic solution to improve system performance. It is used in numerous scientific domains of medical, engineering, and computing, such as manual data entry, automatic spam detection, medical diagnosis, picture identification, data purification, noise reduction, and so on, due to its diverse uses. Recent research shows that machine learning has been used to overcome a variety of problems in WSNs. Using ML in WSNs enhances system performance while also reducing complicated chores such as reprogramming, manually accessing vast amounts of data, and extracting usable data from data. As a result, ML methods are very beneficial for retrieving enormous amounts of data as well as extracting meaningful data [6].

The contribution of this research is as follows:

1. To propose a novel technique in resource allocation (RA) for WSN\_IoT with energy efficiency and data optimization using deep learning architectures;
2. To improve the energy efficiency of the network using a whale-optimization-based deep neural network;
3. To optimize the data transmission of the network using a heuristic-based multi-objective firefly algorithm.

The organization of this article is as follows: Section 2 gives the related works, the proposed technique is described in Section 3, Section 4 explains the performance analysis, and the conclusion is given in Section 5.

## 2. Related Works

ML and DL methods for data processing could make edge devices smarter while also improving privacy and bandwidth usage. The authors of [7] applied deep learning for IoT to an edge computing environment and provided a method for improving network speed while also protecting user privacy. The authors suggested adaptive sampling-based data reduction methodologies in [8]. These methods function by analyzing the level of variance between acquired data over time and dynamically altering the sampling frequency of the sensors. In instances where the gathered time series are stationary, adaptive sampling algorithms function well. These methods perform badly when dealing with rapidly changing data. The authors of [9] developed a dual prediction-based data reduction technique.

The suggested technique works by developing and implementing a model that represents the sensed phenomenon on both the edge node and IoT devices. Prediction techniques have the advantage that the model at the edge predicts the detected measurement without requiring a radio connection until the prediction error exceeds a predetermined threshold. Work [10] demonstrated that man-made consciousness and machine learning can be used to advance to unavoidable systems. AI-assisted ML systems will aid in merging human intuition and ingenuity with AI capacity. AI-powered systems will assist in dynamically analyzing processing scenarios and adapting appropriate scheduling and resource allocation strategies. Ref. [11] suggests a framework for cloud computing systems to increase QoS while lowering the cost of providing services to end users. The system focuses on condensing VMs based on current resource utilization, creating virtual networks between VMs, and dynamically configuring virtual hubs. In WSN-aided IoT, ref. [12] proposed a QoS-aware safe DL technique for dynamic cluster-based routing. In the WSN, the author [13] built a DL link dependability prediction. This research designed a resilient routing algorithm for a better WSN routing mechanism. For lightweight subgraph extraction as well as labelling, a DL method known as Weisfeiler–Lehman kernel and dual convolutional neural network (WL-DCNN) technique is presented. A discussion of RA tactics is available in [14]. They used a multi-target optimization strategy in this research to trade off speed, cost, and availability in a cloud-based application, and their methodology might be up to 20% faster than existing optimization approaches. Their method has been confirmed. The author of [15] proposed a thermal cognizant workload programming strategy to overcome the excessive power consumption and hotness of data centers. They utilized an ANN to predict the data center's heat effect. In [14], a heterogeneous scheduling model is described. Task resource utilization, as observed in the consolidation methods, was not taken into account. Ref. [15] has a download issue with cloud or fog computing. User fairness and the shortest possible delay are ensured by optimizing discharge results and allocating working out resources. The goal of this optimization problem is to reduce the weighted delay and energy consumption expenses. They devised low-complexity, suboptimal methods to address this NP-hard issue. As a result, half-definite relaxation and randomization are used to make discharge decisions, while fractional programming theory is used to manage resources. The authors of [16] provide a heuristic approach to resource allocation, and a TSA is presented. Modules such as divide and conquer TSA and resource allocation, modified analytical processes, the longest projected length of processing, and divisible scheduling with bandwidth knowledge are used in this technique. The tasks are processed before they are assigned. With relation to the load and bandwidth of the cloud holdings, the allocation is performed using BAR optimization and associated BATS algorithms [17] investigates EE for combining BSs as well as beamforming in multicell situations. Ref. [18] investigates the energy and spectrum efficiency of 5G mobile MIMO networks. In [19], the authors developed an energy-efficient non-cooperative game for distributed CRN over interference channels. For CRNs and IoT, a noncooperative game based on power allocation is proposed in [20], which investigates a mesh adaptive search technique for device to device-assisted CRN; [21] proposes a gradient adaptation optimization for power allocation as well as EE in CRNs. Although gradient techniques are reliable, they can fail to accomplish global optimization. Heuristic algorithms are gaining popularity among researchers as a way to lower the computational complexity of optimization approaches. For NP-complete problems, heuristic techniques are simple to use and adapt. In [22], nonorthogonal multiple access (NOMA) is employed for IoT resource management in smart cities, and mixed-integer linear programming is presented for energy harvesting. To maximize EE and SE trade-off in CR-IoT, a mixed-integer nonlinear programming (MINLP) technique is presented. Optimizing and increasing the efficiency of this communication is an important consideration, and resource allocation is a critical bottleneck. Researchers are using innovative AI methods to optimize resource allocation according to the data flow during network operation to solve the challenge of resource allocation. These measures have moved the industry towards automated resource management on a large and complex scale.

### 3. System Model

This section discusses novel techniques in RA for WSN\_IoT with energy efficiency and data optimization using deep learning architectures. Here, EE and SE are considered conflicting optimization objectives. EE of the network is enhanced using a whale-optimization-based deep neural network. Data optimization has been carried out using a heuristic-based multi-objective firefly algorithm.

RA vector  $p(h) \in \mathbb{R}^m$  and a function  $f: \mathbb{R}^m \times \mathbb{R}^n \rightarrow \mathbb{R}^u$  are associated with each fading channel realization. When the channel realization is  $h$ , the components of vector valued function  $f \cdot p(h)$ ,  $h$  relates performance indicators linked with resource allocation  $p(h)$ . The system allocates resources instantly in fast time changing fading channels, and consumers. This supports examining vector ergodic average  $x = E \cdot f \cdot p(h)$ ,  $h \in \mathbb{R}^u$ , which is relaxed to the inequality for defining optimal wireless design issues by Equation (1).

$$x \leq \mathbb{E}[f(p(h), h)] \quad (1)$$

At ideal operating points, we will obtain  $x = E \cdot f \cdot p(h)$ ,  $h$ , but this is not required a priori. In optimally designed WCS, the goal is to identify instantaneous RA  $p(h)$  that in some way improves performance metric  $x$ .  $P$  has bounded functions, implying that resources being allocated are finite. With these definitions, we can write a software to solve optimal RA problems in WCN by Equation (2).

$$\begin{aligned} P^* &:= \max_{p(h), x} g_0(x) \\ \text{s.t. } x &\leq \mathbb{E}[f(p(h), h)] \\ g(x) &\geq 0, x \in \mathcal{X}, p \in \mathcal{P} \end{aligned} \quad (2)$$

Because (5) requires RA functions to follow parametrization  $p(h) = \varphi(h, \theta)$ , this results in a loss of optimality. We focus our attention in this paper on a frequently utilized family of parameterizations known as near universal, which may represent any function in  $P$  to a specified precision.

Energy optimization using whale-optimization-based deep neural network:

There are two stages to the optimization process. The spiral position is updated in the first step, and the prey is circled. The second stage involves a random search for prey. Following is a mathematical representation of the stated step.

Encircling prey: Whales track down their prey and surround them. The position of prey in the search space is unknown. According to the WOA, the leading factor is an ideal prey. The surviving search agents aim to improve their performance by switching locations. The following are examples of search agent behaviors by Equation (3):

$$\begin{aligned} \vec{Y}(u+1) &= \vec{Y}^*(u) - \vec{B} \cdot \vec{E} \\ \vec{E} &= \left| \vec{C} \cdot \vec{Y}^2(u) - \vec{Y}(u) \right| \end{aligned} \quad (3)$$

after iteration  $u$ ,  $\vec{Y}^2(u)$  defines the whale's ideal location. The whale's current location is  $\vec{Y} \cdot u + 1$ , and a distance vector represents the distance between whale and prey. The absolute value is represented by  $||$ .  $B$  and  $C$  coefficient vectors are calculated as follows by Equation (4):

$$\begin{aligned} \vec{B} &= 2 \cdot \vec{b} \cdot \vec{s} + \vec{b} \\ \vec{C} &= 2 \cdot \vec{s} \end{aligned} \quad (4)$$

The value of  $b$  is minimized in order to apply to shrink, and  $B$ 's oscillation range is reduced to. The value of  $B$  is lowered from 2 to 0 iterations, while the value of  $b$  is minimized from 2 to 0. The best agent's position as well as an agent's initial location are decided by choosing a random value of  $B$  between (1, 1).

#### A. Spiral position updating

Calculating the distance between whale location  $Y, Z$  and prey location  $Y, Z$  defines a helix shape for whale prey tracking. The movement toward prey is described as by Equation (5)

$$\vec{E} = \left| \vec{Y}^2(u) - \vec{Y}(u) \right| \quad (5)$$

The constant  $b$  identifies the curvature of the logarithmic spiral, and random numbers vary from 1 to 1. Whales can shift their location while undertaking reduction thanks to their circular movement. Between spiral and diminishing encircling, there is a 50% chance of selection by Equation (6)

$$\vec{Y}(u+1) = Y_{\text{rand}} - \vec{B}, \vec{E} \quad (6)$$

#### B. Prey search

The exploration step, which is dependent on vector variations, is also known as the prey hunting stage. The whale conducts a random search for the prey based on its position. The search agent flees the seeking whale due to the whale's location. WOA uses the vector  $B$ , which has random values less than or equal to 1. The search agent is chosen at random during the exploration phase. By lowering the local optimization problem, random selection makes WOA a GSA. The following is definition of a global search by Equation (7):

$$\begin{aligned} \vec{Y}(u+1) &= Y_{\text{rand}} - \vec{B}, \vec{E} \\ \vec{E} &= \left| \vec{C} \cdot Y_{\text{rand}} - \vec{Y} \right| \end{aligned} \quad (7)$$

where  $Y_{\text{rand}}$  is a randomly chosen whale from the specified population. The whale population is given random outcomes by the WOA, which assumes the optimal solution of function. Fitness function analyses selected features from this phase and found that a few are still redundant, affecting accuracy of final classification. The fitness function is an ESD classifier, and the error is evaluated for every iteration. As a result, we established a new phase called "additional features approval" (EFA). The standard error of the mean is used in this step (SEM). For the final recognition, the SEM value is put through a threshold function. This function is described by Equation (8):

$$\begin{aligned} \text{SEM}(y_i) &= \frac{\sigma}{N}, \sigma = \sqrt{\sigma^2}, \sigma^2 = E[(y_i - \mu)^2] \\ \text{Selection} = F(i) &= \begin{cases} F_i^s & \text{for } y_i \geq \text{SEM}(y_i) \\ \text{Remove} & \text{Elsewhere} \end{cases} \end{aligned} \quad (8)$$

The final recognition is performed using an ensemble subspace classifier (Algorithm 1).

**Algorithm 1** The whale optimization algorithm

---

```

1. Initial population  $Y_i$  where  $(i = 1, 2, 3, \dots, n)$ 
2. Fitness evaluation for every solution
3.  $Y^* =$  best search agent
4. For each solution
5.   While ( $u < \text{Max\_iteration}$ )
6.   Updated  $b, B, C, M$  and  $p$ 
7.   If 1 ( $p < 0.5$ )
8.   If 2 ( $B/ < 1$ )
9.     Random search agent selection ( $Y_{\text{rand}}$ )
10.  Else if ( $B/ > 1$ )
11.    Current search agent location changes.
12.  End if
13. Else if 1 ( $p \geq 0.5$ )
14. End While
15.   Change  $Y^*$  if better solution is accessible  $u = u + 1$ 
16. Return  $Y^*$ 
17. Output:  $F_i^s \leftarrow$  Best Feature Vector

```

---

We used the whale optimization algorithm in our research to determine the ideal set of parameters for DNN method training as well as hyperparameter selection. We establish three things to specify our problem. To ensure the easy replication of our results, we kept the initial divides into train and test subsets throughout our studies. The creators of the dataset submitted an unmodified, fixed test set, which was used to evaluate the final results. We used a simple DL architecture built from fully connected layers to handle these basic tasks. We used the following layers to train the network for picture classification:

- Flatten layer to resize images;
- 512 neurons in a dense (fully linked) layer with ReLU activation function;
- To categorize 10 classes, use a dense layer with 10 neurons and the Softmax activation function.

The following layers make up the network utilised for Reuters dataset:

- Two dense layers, each with 64 neurons, with ReLU activation;
- Softmax activation function for 46 classes and one dense layer with 46 neurons. The steps in a typical deep learning process are as follows: (1) data preprocessing, (2) DNN method implementation and configuration, (3) learning as well as fitting method to data. The cost function is used heavily in the final stage since its behavior is determined by a set of training parameters. Only training parameters are optimized. Our search area is an  $n$ -dimensional grid, with  $n$  equaling the number of specifications. The algorithm was then tweaked to work in three dimensions, and three parameters were optimized. The approach can possibly be extended to  $n$  dimensions, as stated in [8]. To do so, we refactored the solution vectors as well as the entire mathematical method to operate in three dimensions. Our 3D-WOA method was based on original open source one, but with the addition of 3D vectors. Finally, APIs for runtime as well as benchmark functions are updated to support three-dimensional space.

To begin, a separate class with an optimization function is described to optimize the cost function. It accepts training parameters as arguments and performs a fitting operation on them. We decided to optimize solely on cross-validation accuracy metrics, which function returns as well as serves as a cost function value. The specifications that control the search space and constraints must be properly modified while utilising the whale optimisation algorithm. The issue with the technique was that WOA only operates in continuous space, whereas the specifications of NN have discrete values. As a result, the WOA had to be modified to handle discrete optimization problems. To do this, discretization was carried out within the algorithm class's main body. At the time as solution vector formation, all parameters were rounded to the nearest integer; fresh solutions produced by WOA were instantly converted to nearest integer as well as returned

to main body of method as discrete values. Furthermore, lookup table encoding has been employed in some circumstances, such as for coding optimizer selections. Furthermore, we move through a search space appropriate for a specific issue by adding appropriate restrictions to the solutions. The first argument determines number of iterations to execute, while second gives number of agents. Both specifications have a significant impact on method output: the higher they are, the more swarm-based the method becomes. Each successive generation or agent necessitates the training of a deep neural network from the ground up. The default values for parameters  $a$  and  $b$  have been specified.

#### C. Heuristic based multi-objective firefly algorithm

One method for multiobjective optimization is to integrate all objectives into a single goal, allowing single-objective optimization techniques to be employed without significant change (Algorithm 2). FA are utilized to solve multiobjective issues directly in this way. Another option is to expand the firefly technique to directly produce Pareto optimum fronts. We may construct MOFA by extending core ideas of FA.

---

#### Algorithm 2 The multiobjective optimization algorithm

---

1. Describe objective functions  $f_1(x), \dots, f_K(x)$  where  $x = (x_1, \dots, x_d)^T$
  2. **while** ( $t < \text{MaxGeneration}$ )
  3. Start a population of  $n$  fireflies  $x_i (i = 1, 2, \dots, n)$
  4. **for**  $i, j = 1 : n$  (all  $n$  fireflies)
  5. Calculate their approximations  $PF_i$  and  $PF_j$  to Pareto front
  6. **if**  $PF_j$  dominates  $PF_i$ ,
  7. **if**  $i \neq j$  and when all constraints are gratified
  8. Move firefly  $i$  towards  $j$  using (2)
  9. **if** no non-dominated solutions are found
  10. Produce new ones if moves do not satisfy all constraints **end if**
  11. Determine best solution  $g_*^t$  to reduce  $\psi$  in (4)
  12. Produce random weights  $w_k (k = 1, \dots, K)$
  13. Random walk around  $g_*^t$
  14. **end if**
  15. Sort and determine current best approximation to Pareto front
  16. Update  $t \leftarrow t + 1$
  17. **end while**
- 

#### D. Postprocess results and visualization

The method starts with the correct specification of goal functions as well as nonlinear constraints. After that, a random weight vector is constructed in order to obtain a combined best solution  $g \cdot t$ . Answers that are not dominated are then passed on to the next iteration. In general,  $n$  non-dominated solution locations are obtained after a certain number of repetitions to method genuine Pareto front. We can identify the current best  $g \cdot t$ , which reduces a combined objective to execute random walks more efficiently by Equation (9):

$$\psi(x) = \sum_{k=1}^K w_k f_k, \sum_{k=1}^K w_k = 1 \quad (9)$$

$$w_k = p_k / K$$

where  $p_k$  is a set of random numbers selected from a uniformly distributed Unif [0, 1]. After creating  $K$  uniformly distributed numbers, a rescaling procedure is undertaken to ensure that  $P_k \cdot w_k = 1$ . It is worth noting that the weights  $w_k$  should be picked at random by Equation (10).

$$x_i^{t+1} = g_*^t + \alpha_t \in_i^t \quad (10)$$

In iteration progress, randomness is minimized in a way similar to that of simulated annealing and other random reduction methods [11].

$$\alpha_t = \alpha_0 0.9^t \quad (11)$$



where  $\alpha_0$  is the initial randomness factor.

$$u \preceq v \iff u \prec v \vee u = v \quad (12)$$

It is worth noting that the dominance may be defined for maximization issues by replacing  $\prec$  with  $\succ$ . As a result, if no solution dominates a point  $x$ , it is called a non-dominated solution [5]. The set of non-dominated solutions in a multiobjective Pareto front P·F can be described as Equation (13):

$$PF = \{s \in \mathcal{S} \mid \nexists s' \in \mathcal{S} : s' \prec s\} \quad (13)$$

where  $\mathcal{S}$  is solution set. A variety of solutions should be created utilizing efficient strategies. As we may see from later simulations, Levy flights, for example, ensure a good diversity of options.

When  $x_i$  approaches, the discrete components of  $x_i$  are more likely to convert from binary to real numbers to convert a real number into a binary one by Equation (14):

$$\text{sig}(x_l^i) = \frac{1}{1 + \exp(-x_l^i)} \quad (14)$$

where  $x_l^i$  is component  $l$  of position vector  $x_i$  of firefly  $I$ , following migration in the context of FA—recall (7) and (4). Equation (9) represents a solution's floating-point components as a set of probabilities. The proper binary values are then assigned using Equation (15).

$$b_l^i = \begin{cases} 1, & \text{if rand} \leq \text{sig}(x_l^i) \\ 0, & \text{otherwise,} \end{cases} \quad (15)$$

where  $\text{sig}(x_l^i)$  is the likelihood that a component is 0 or 1, and  $\text{rand} \in [0, 1]$  is the probability that the component is 0 or 1. The firefly positions,  $x$ , were not permitted to shift beyond the search space  $\Omega$  during the iterative procedure.

Error function is a specific shape function that is mostly used in probability and statistics. The mathematical function described by an integral is denoted by “erf” in Equation (16)

$$\text{erf}(t) = \frac{2}{\sqrt{\pi}} \int_0^t \exp(-t^2) dt \quad (16)$$

which satisfies the following properties:

$$\text{erf}(0) = 0, \text{erf}(-x) = -\text{erf}(x) \quad (17)$$

The erf function's derivative follows directly from its definition by Equation (18):

$$\frac{d}{dt} \text{erf}(t) = \frac{2}{\sqrt{\pi}} \exp(-t^2), \text{ for } t \in \mathbb{R}. \quad (18)$$

Thus, the erf function's favorable features are utilized to describe a new sigmoid function, sigmoid erf function by Equation (19):

$$\text{sig}(x_l^i) = 0.5 \left( 1 + \text{erf}\left(\frac{x_l^i}{\sqrt{2}}\right) \right) \quad (19)$$

which is a bounded differentiable real function with a positive derivative at every point and defined for any  $x \in \mathbb{R}$ . Note that the sigmoid function in (14) has a slope of about 0.5641895 at the origin, whereas Equation (9) has a slope of 0.25, resulting in a faster rise from 0 to 1. The simplest discretization process is for the floor function, which rounds to the integer



part. For  $l = 1, \dots, n$ , each continuous value  $x_{il} \in \mathbb{R}$  is changed into a binary one, 0 bit or 1 bit, in the following way by Equation (20):

$$b_l^i = \left\lfloor \left\lfloor x_{il} \bmod 2 \right\rfloor \right\rfloor \quad (20)$$

where  $\lfloor z \rfloor$  is the  $z$  floor function and returns the greatest integer not exceeding  $z$ . The absolute value of the remainder is floored after the floating-point value  $x_{il}$  is divided by 2.

The movement of every firefly is made on continuous space in this implementation of previously mentioned heuristics, indicated by “movement on continuous space” (mCS), using true location vector. Only after all motions toward brighter fireflies are completed is the real position of firefly  $i$  discretized. The fitness evaluation of each firefly is always dependent on the binary position for firefly ranking.

#### 4. Performance Analysis

The performance of proposed DL routing protocol is calculated, utilizing the MATLAB platform to simulate the suggested architecture. The number of nodes used in this experiment might range from 200 to 1000. These nodes are spread out over a  $1000 \times 1000 \text{ m}^2$  region. The proposed WODNN HMOFA is compared against five current algorithms, including the GEEC protocol, TTDFP, and EADCR. Table 1 lists the simulated parameters used in this study.

**Table 1.** Simulation Parameters.

Parameter	Value(s)
Number of SNs	200 to 1000
Initial energy	0.25 nJ
Sensor field	10,001,000
Transmission energy	50 nJ/bit
Multipath (amplification)	0.0013 pJ/bit/m <sup>4</sup>
Free space	10 nJ/bit <sup>2</sup> m <sup>2</sup>
Data packet size	4000 bits
Effective data aggregation	5 nJ/bit/signal
Threshold distance	87 m
Absolute remaining energy	0.2
CH selection probability	0.1

The above Tables 2–6 shows comparative analysis of WSN based resource allocation using deep learning in energy efficiency with data optimization. Here, parametric analysis is carried out in terms of throughput, energy efficiency, QoS, throughput, and spectral efficiency. The proposed technique obtained optimal results as throughput of 96%, energy efficiency of 95%, QoS of 75%, spectral efficiency of 85%, and network lifetime of 91%, as shown in Figures 1–5. From the above analysis, the energy efficiency and data optimization have been optimized by the proposed technique.

**Table 2.** Parametric analysis of throughput.

Number of Nodes	WL_DCNN	NOMA	RA_WSN_IoT
200	65	69	72
400	69	73	77
600	71	75	82
800	73	79	89
1000	75	83	96

**Table 3.** Parametric analysis of energy efficiency.

Number of Nodes	WL_DCNN	NOMA	RA_WSN_IoT
200	71	75	79
400	73	79	82
600	75	83	85
800	79	85	90
1000	82	86	95

**Table 4.** Parametric analysis of QoS.

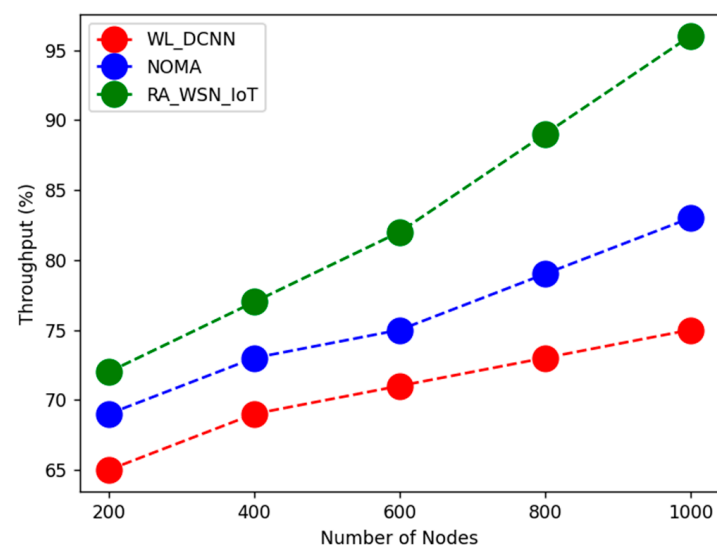
Number of Nodes	WL_DCNN	NOMA	RA_WSN_IoT
200	55	59	63
400	59	62	65
600	63	65	69
800	65	69	72
1000	69	73	75

**Table 5.** Parametric analysis of spectral efficiency.

Number of Nodes	WL_DCNN	NOMA	RA_WSN_IoT
200	59	62	69
400	62	68	73
600	65	72	75
800	69	75	79
1000	73	79	85

**Table 6.** Parametric analysis of network lifetime.

Number of Nodes	WL_DCNN	NOMA	RA_WSN_IoT
200	61	65	69
400	65	69	75
600	66	73	79
800	69	75	85
1000	72	79	91

**Figure 1.** Comparative analysis of throughput.

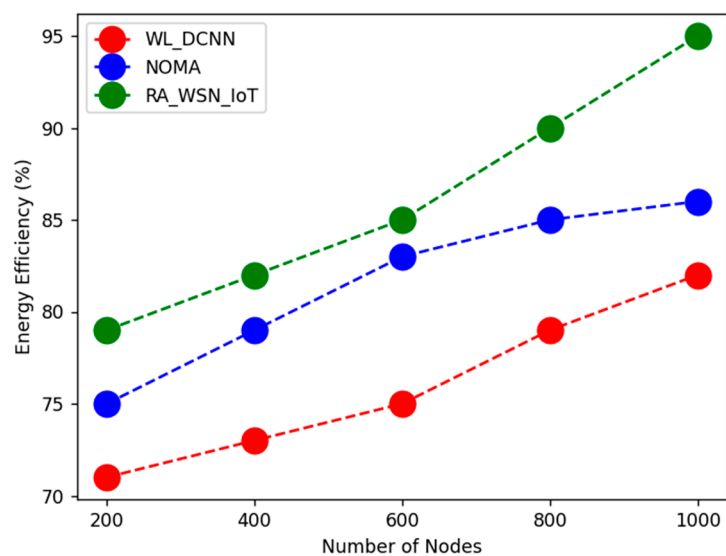


Figure 2. Comparative analysis of energy efficiency.

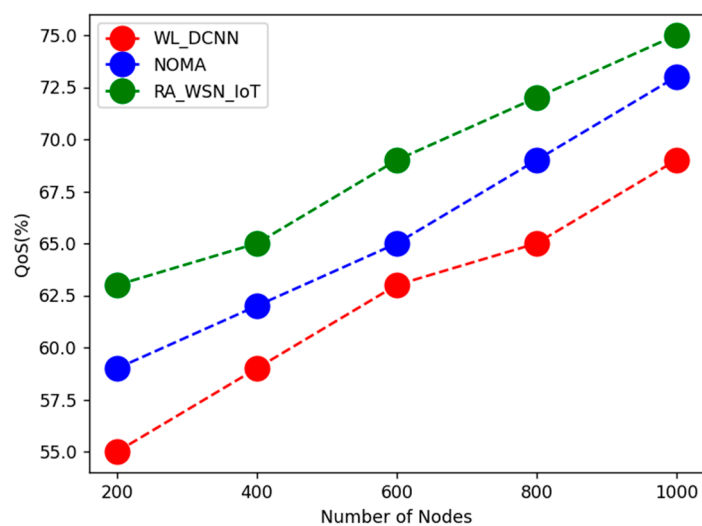


Figure 3. Comparative analysis of QoS.

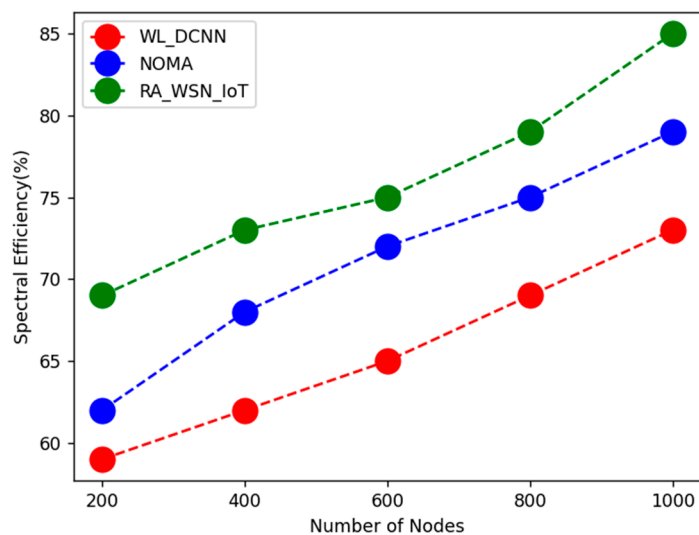
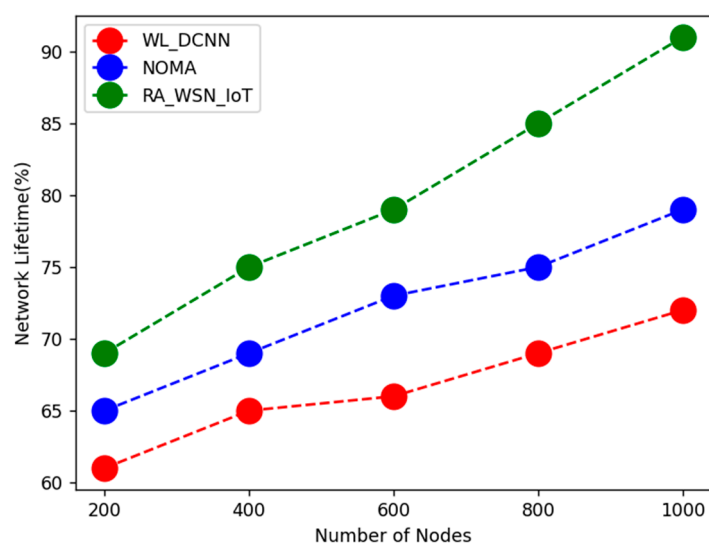


Figure 4. Comparative analysis of spectral efficiency.



**Figure 5.** Comparative analysis of network lifetime.

## 5. Conclusions

This study uses deep learning architectures to provide a unique resource allocation strategy for WSN\_IoT with EE and data optimization. EE and SE are two competing optimization goals in this case. The network's energy efficiency has been improved by a DNN based on whale optimization. The heuristic-based multi-objective firefly algorithm was used to optimize the data. This proposed method is applied to optimal power allocation and relay selection. The study is for a cooperative multi-hop network topology. The best resource allocation is achieved by reducing overall transmit power, and the best relay selection is accomplished by meeting QoS standards. As a result, an energy-efficient protocol has been achieved. The simulation results demonstrated the suggested model's competitive performance when compared to traditional models in terms of throughput of 96%, energy efficiency of 95%, QoS of 75%, spectrum efficiency of 85%, and network lifetime of 91 percent. The future scope of this research can be used for medical application with enhanced security in real-time sensor-based techniques, which can enhance the efficiency of the proposed technique using a machine learning technique integrated with blockchain techniques.

**Author Contributions:** Conceptualization, Q.W.A., S.G., A.R., M.R., N.Z.J., M.M. and M.B.; methodology, Q.W.A., S.G., A.R., M.R., N.Z.J., M.M. and M.B.; software, Q.W.A., S.G., A.R., M.R., N.Z.J., M.M. and M.B.; validation, Q.W.A., S.G., A.R., M.R., N.Z.J., M.M. and M.B.; formal analysis, Q.W.A., S.G., A.R., M.R., N.Z.J., M.M. and M.B.; investigation, Q.W.A., S.G., A.R., M.R., N.Z.J., M.M. and M.B.; resources, Q.W.A., S.G., A.R., M.R., N.Z.J., M.M. and M.B.; data curation, Q.W.A., S.G., A.R., M.R., N.Z.J., M.M. and M.B.; writing—original draft preparation, Q.W.A., S.G., A.R., M.R., N.Z.J., M.M. and M.B.; writing—review and editing, Q.W.A., S.G., A.R., M.R., N.Z.J., M.M. and M.B.; visualization, Q.W.A., S.G., A.R., M.R., N.Z.J., M.M. and M.B.; supervision, N.Z.J.; funding acquisition, Q.W.A., S.G., A.R., M.R., N.Z.J., M.M. and M.B. All authors have read and agreed to the published version of the manuscript.

**Funding:** Authors would like to thank for the support of Taif University Researchers Supporting Project number (TURSP-2020/239), Taif University, Taif, Saudi Arabia.

**Data Availability Statement:** Data will be shared for review based on the editorial reviewer's request.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

- Reddy, K.H.K.; Luhach, A.K.; Pradhan, B.; Dash, J.K.; Roy, D.S. A genetic algorithm for energy efficient fog layer resource management in context-aware smart cities. *Sustain. Cities Soc.* **2020**, *63*, 102428. [\[CrossRef\]](#)
- Nayak, P.; Swetha, G.K.; Gupta, S.; Madhavi, K. Routing in wireless sensor networks using machine learning techniques: Challenges and opportunities. *Measurement* **2021**, *178*, 108974. [\[CrossRef\]](#)
- Mohanty, S.N.; Lydia, E.L.; Elhoseny, M.; Al Otaibi, M.M.G.; Shankar, K. Deep learning with LSTM based distributed data mining model for energy efficient wireless sensor networks. *Phys. Commun.* **2020**, *40*, 101097. [\[CrossRef\]](#)
- Zhai, D.; Wang, C.; Zhang, R.; Haotong, C.; Yu, F.R. Energy-Saving Deployment Optimization and Resource Management for UAV-Assisted Wireless Sensor Networks with NOMA. *IEEE Trans. Veh. Technol.* **2022**. [\[CrossRef\]](#)
- Ali Imran, M.; Flávia dos Reis, A.; Brante, G.; Valente Klaine, P.; Demo Souza, R. Machine learning in energy efficiency optimization. *Mach. Learn. Future Wirel. Commun.* **2020**, 105–117. [\[CrossRef\]](#)
- Mukherjee, A.; Goswami, P.; Khan, M.A.; Manman, L.; Yang, L.; Pillai, P. Energy-efficient resource allocation strategy in massive IoT for industrial 6G applications. *IEEE Internet Things J.* **2020**, *8*, 5194–5201. [\[CrossRef\]](#)
- Radhika, S.; Rangarajan, P. Fuzzy based sleep scheduling algorithm with machine learning techniques to en-hance energy efficiency in wireless sensor networks. *Wirel. Pers. Commun.* **2021**, *118*, 3025–3044. [\[CrossRef\]](#)
- Gulganwa, P.; Jain, S. EES-WCA: Energy efficient and secure weighted clustering for WSN using machine learn-ing approach. *Int. J. Inf. Technol.* **2022**, *14*, 135–144. [\[CrossRef\]](#)
- Tyagi SK, S.; Mukherjee, A.; Pokhrel, S.R.; Hiran, K.K. An intelligent and optimal resource allocation ap-proach in sensor networks for smart agri-IoT. *IEEE Sens. J.* **2020**, *21*, 17439–17446. [\[CrossRef\]](#)
- Kaur, G.; Chanak, P.; Bhattacharya, M. Energy-efficient intelligent routing scheme for IoT-enabled WSNs. *IEEE Internet Things J.* **2021**, *8*, 11440–11449. [\[CrossRef\]](#)
- Wang, W.; Guo, H.; Li, X.; Tang, S.; Xia, J.; Lv, Z. Deep learning for assessment of environmental satisfaction using BIM big data in energy efficient building digital twins. *Sustain. Energy Technol. Assess.* **2022**, *50*, 101897. [\[CrossRef\]](#)
- Goswami, P.; Mukherjee, A.; Maiti, M.; Tyagi SK, S.; Yang, L. A neural network based optimal resource allocation method for secure IIoT network. *IEEE Internet Things J.* **2021**, *9*, 2538–2544. [\[CrossRef\]](#)
- John, N.M.; Joseph, N.; Manuel, N.; Emmanuel, S.; Kurian, S.M. Energy efficient data aggregation and improved prediction in cooperative surveillance system through Machine Learning and Particle Swarm based Optimization. *EAI Endorsed Trans. Energy Web* **2021**, e40. [\[CrossRef\]](#)
- Goyal, N.; Sandhu, J.K.; Verma, L. Machine learning based data agglomeration in underwater wireless sensor networks. *Int. J. Manag. Technol. Eng.* **2019**, *9*, 240–245.
- Puri, D.; Bhushan, B. Enhancement of security and energy efficiency in WSNs: Machine Learning to the rescue. In Proceedings of the 2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), Greater Noida, India, 18–19 October 2019; pp. 120–125.
- Abdul-Qawy, A.S.H.; Almurisi, N.M.S.; Tadisetty, S. Classification of energy saving techniques for IoT-based heterogeneous wireless nodes. *Procedia Comput. Sci.* **2020**, *171*, 2590–2599. [\[CrossRef\]](#)
- Quasim, M.T. Resource management and task scheduling for IoT using mobile edge computing. *Wirel. Pers. Commun.* **2021**, 1–18. [\[CrossRef\]](#)
- Soundari, A.G.; Jyothi, V.L. Energy efficient machine learning technique for smart data collection in wireless sensor networks. *Circuits, Syst. Signal Process.* **2020**, *39*, 1089–1122. [\[CrossRef\]](#)
- Ding, Q.; Zhu, R.; Liu, H.; Ma, M. An overview of machine learning-based energy-efficient routing algorithms in wireless sensor networks. *Electronics* **2021**, *10*, 1539. [\[CrossRef\]](#)
- Guo, Y.-Y.; Yang, J.; Tan, X.-L.; Liu, Q. An Energy-Efficiency Multi-Relay Selection and Power Allocation Based on Deep Neural Network for Amplify-and-Forward Cooperative Transmission. *IEEE Wirel. Commun. Lett.* **2021**, *11*, 63–66. [\[CrossRef\]](#)
- Goswami, P.; Mukherjee, A.; Hazra, R.; Yang, L.; Ghosh, U.; Qi, Y.; Wang, H. AI based energy efficient routing protocol for intelligent transportation system. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 1670–1679. [\[CrossRef\]](#)
- Zhao, B.; Zhao, X. Deep Reinforcement Learning Resource Allocation in Wireless Sensor Networks with Energy Harvesting and Relay. *IEEE Internet Things J.* **2021**, *9*, 2330–2345. [\[CrossRef\]](#)