Hindawi
Mathematical Problems in Engineering
Volume 2023, Article ID 9937949, 9 pages
https://doi.org/10.1155/2023/9937949

Research Article

An Energy Efficient Evolutionary Approach for Smart City-Based IoT Applications

Rashmi Prava Das,[1] Tushar Kanta Samal,[1] and Ashish Kr. Luhach[2]

[1]*Department of Computer Science and Engineering, CV Raman Global University, Bhubaneswar, India*
[2]*The Papua New Guinea University of Technology, Lae, Papua New Guinea*

Correspondence should be addressed to Ashish Kr. Luhach; ashish.kumar@pnguot.ac.pg

Received 20 February 2023; Revised 6 November 2023; Accepted 11 November 2023; Published 7 December 2023

Academic Editor: Anna M. Gil-Lafuente

Internet of Things (IoT) has been used in smart cities, agriculture, weather forecasting, smart grids, and waste management. The IoT has huge potential but needs refinement. The paper focuses on lowering IoT sensor power consumption to improve network life. This work selects the best IoT cluster header (CH) to maximize energy consumption. The suggested technique uses particle swarm optimization (PSO) with artificial neural networks (ANNs). The optimal CH in an IoT network cluster was identified by taking into account the number of active nodes, the load, the residual energy, and the cost function. This work compares the suggested method with artificial bee colony, genetic, and adaptive gravity search algorithms. The hybrid solution beats conventional methods.

1. Introduction

A smart city is an urban environment that uses information and communication technology (ICT) and other closely related technologies to improve the efficiency of everyday urban activities and the quality of service (QoS) provided to urban residents [1, 2]. Continued population growth and urbanization are driving the need for creative approaches to urban management with minimal impact on the environment, public life, and governance. The process of integrating intelligent urbanization has been facilitated through the use of ICT in city operations. With the proliferation of smart devices and recent advances, the idea of smart cities has gained a lot of attention. Internet of Things (IoT) has been used in smart cities, agriculture, weather forecasting, smart grids, and waste management etc [3, 4]. As a result of the Internet of Things (IoT), traditional networks have emerged, connecting millions of connected devices. Further, realization of this IoT concept in smart cities requires continuous technological advances in ubiquitous computing, wireless sensor networks (WSNs), machine-to-machine communication, and design of cross-industry applications.

The de facto policy of the IoT is to enable unified communications through uniquely identifiable smart devices with minimal human intervention. The IoT paradigm has attracted significant interest from a wide variety of interest groups, including smart city, smart home, smart health, and many others.

Getting everything from the cloud is getting more complicated. Cloud computing provides a variety of resources, such as storage and processing, that you can access when you need them. Despite its many advantages, there are also some drawbacks that prevent full adoption of the cloud and realization of its various advantages. The problems we faced with cloud infrastructure adaptability were high latency, high power consumption, and uncertain geographic location.

To address these issues, the idea of fog computing was developed to improve and ensure real-time responsiveness while minimizing cloud load [5]. End users and cloud data centers are usually separated by a layer of fog. Edge detection, location detection, geographical edge adaptation, wireless network access, and power reduction are the main functions of this core. In most cases, it is not practical to exceed the amount of power consumed by a cloud data center. Some energy consumption can be minimized, but the fog layer uses small self-powered data centers to make minor adjustments to minimize power consumption. Fog computing can help manage smart devices by adding data to the network's edge. Fog is context aware, low latency, and mobile. As IoT devices

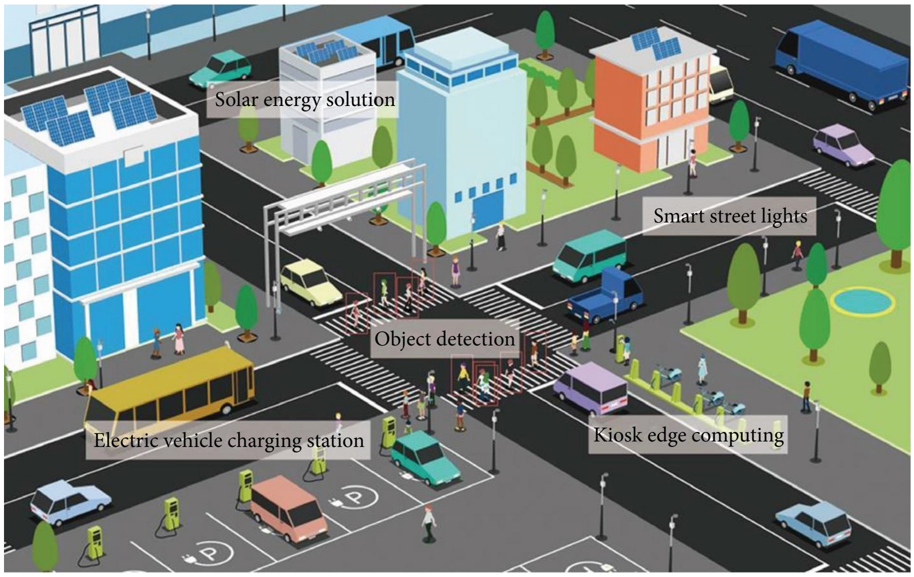

Figure 1: A complex smart city scenario with ICT infrastructure conceptual layers.

proliferate, cloud utilization increases. Due to fog's popularity, researchers monitor energy use. Figure 1 shows the detailed layout of the access point and fog gateway, and section Abbreviations describe the abbreviation used in this research work. Using context-aware computing at the 45-fog layer has been proven successful in processing data streams from various IoT applications [6–8].

The fog layer computings are intended to offload tasks from the cloud layer and increase service response time [9, 10]. As sustainability is one of the most pressing issues in smart infrastructure today, energy conservation at all levels of smart city IoT infrastructure is an important research work [11, 12].

Numerous research projects have been conducted on the topic of energy-saving strategies that can be found in the device or sensor layer [13–15]. Duty cycle mechanisms play an important role in reducing energy consumption at both system and network levels in IoT environments [16–19]. Dhall and Agrawal [16] introduced an improved duty cycling algorithm that uses effective path selection methods based on residual energy parameters to minimize power consumption and maximize network reliability and lifetime. Amirtharaj et al. [17] proposed an algorithm to improve the duty cycle efficiency of Linux-based IoT devices by finding and deactivating units that take up time and aren't necessary that are initialized in user space. It presents an energy-efficient context-aware traffic scheduling method that first defines IoT applications based on the diverse traffic requirements they have and then assigns those apps to various weighted quality classes [18]. Kozłowski and Sosnowski [19] proposed various power consumption models for the various duty cycle schemes and the maximum power level consumed by the additional wake-up system was determined to determine commonly used duty cycles in typical IoT networks that made it a fair alternative to the cycle method. IoT and WSN environments typically consist of large numbers of low-cost, battery-powered devices with limited functionality. Because IoT devices are frequently installed in hospitable and unmanned environments, it is impractical to change their batteries. Therefore, the most critical aspect in the process of creating an effective IoT or WSN is the conservation of power [7, 8, 20–22].

Early WSN network research focused on smart grids and environmental and agricultural monitoring, utilizing low-rate, delay-tolerant techniques. Smart cities, industrial automation, healthcare, transit automation, and multimedia use IoT technologies [7, 8]. These are classified as delay-sensitive applications and must be considered for timing constraints. Therefore, the second most important issue in IoT is node-to-node and end-to-end latency. Furthermore, it is difficult to ensure a given delay [20–22]. In different areas of IoT and WSN, researchers have proposed different energy harvesting approaches [21, 23–25] to meet the demand of different energy sources. The models proposed by Barath et al. [26], Zhan and Lai [27], and Sendra Compte et al. [28] are useful for predicting potential energy demand based on historical evidence. On the other hand, if the harvest is uncontrolled or unpredictable, this algorithm will not work well. Various models have been proposed to overcome this limitation by Beheshtiha et al. [29], Briante et al. [30], Oliveira and Castro [31], and Oliveira et al. [32], with the aim of balancing power supplies and minimizing dispersion at different sensor nodes under different environmental conditions. Various update algorithms are used. A number of references [33, 34] often deal with energy minimization in cloud layers.

Fog devices are commonly employed in IoT contexts, and study shows fog node middleware has more computational capacity than end devices but less than cloud centers. Ma et al. [35] proposed this IoT-based fog computing model that used genetic algorithms to reduce faulty nodes and power consumption. IoT networks are efficiently handled using fog nodes, edge nodes, or a combination of cloud and fog by Skarlat et al. [36], Souza et al. [37], Minh et al. [38], Naas et al. [39], Taneja and Davy [40], and Lera et al. [41].

Various service deployment methods have been proposed for different platforms to reduce power consumption. Similarly, there are many publications where authors used different optimization strategies to reduce energy consumption at fog and edge levels. In contrast, advances in fog layer research in smart city research have only recently been recognized for their relevance and research potential [42–44]. Examples of such applications of scenarios include large-scale video summarization for smart cities [42], urban traffic monitoring [43], and automatic adjustment of surveillance cameras for smart cities [44]. However, little research has focused on the energy savings of fog layers. To further enhance the effectiveness of performance mediation, some authors apply different optimization strategies to optimize performance mediation at different levels [45–49]. An optimized model based on genetic algorithms, proposed in a study by Skarlat et al. [45], considers application and resource QoS heterogeneity to install this IoT application using fog resources. Zubair et al. [46] and Sun et al. [47] used a multiobjective optimization technique to solve the placement of services on different platforms. Guerrero et al. [48] gave a comparative overview of different optimization techniques. Kaur and Sood [49] presented forecasting and predictive models that solve work scheduling problems using artificial neural networks and genetic algorithms.

Traffic, load, and temperature make IoT increasingly energy intensive. Therefore, a high-energy-efficiency IoT architecture must be proposed to ensure network stability and lifespan. For this purpose, this study uses a hybrid particle swarm optimization-neural network (PSO-NN) to find the optimal cluster header (CH) in specific IoT networks. This hybrid strategy begins with a search for the most significant regions that have been uncovered by the PSO algorithm. Next, the NN is used to increase utilization of the found regions.

Listed below are some of the primary contributions made by the hybrid model that was suggested.

(1) Select the optimal CH using the PSO hybrid model.
(2) Utilize the PSO-NN model in order to improve the performance metrics of WSN sensors, namely those that have an impact on the amount of energy consumed, such as load, temperature, delay, and distance between BT and CH.

The rest is structured like this: The most recent research on how to optimize the power consumption of WSN sensors in IoT systems is discussed in detail in Section 2. Section 3 gives architectural context. Section 4 outlines the design proposal. Section 5 analyzes experimental data. Section 6 discusses conclusions and future study.

2. Related Works

IoT networks have been expanded in several ways. This section examines renowned works. Neuro-fuzzy rule clustering improves end-to-end efficiency and packet delivery rate. This technique helps forward packets effectively and extends network life [11]. Calculating CH's leftover energy and distance to the sink node offer network efficiency. Simulations suggest that neuro-fuzzy reduces energy consumption and increases network longevity. The creators of this work believe that all network nodes can be trusted; however, this isn't necessarily true.

A methodology for choosing the best cluster header (CH) is also suggested. Short battery life, memory, and communication distances characterize sensor nodes. This communication protocol conveyed the authors' data to IoT BT [12]. Two of the best CHs within the same cluster are selected during the CH selection process to extend the network lifetime. The method of choosing dual CH is data entropy knowledge fusion. Entropy data are used for classification and fusion. Dinesh Reddy et al. [13] combined moth flame optimization (MFO) and ant lion optimization (ALO) to choose the ideal CH in WSN-IoT networks. This methodology selects the best CH by choosing the node nearest to the BT, transmits data quickly, saves node power, and reduces IoT device load and temperature. ABC, AGSA, PSO, GA, ALO, GSA, and MFO were compared to the hybrid model. Modern hybrid models store more WSN-IoT resources. It constructs an energy-aware CH selection model for WSN-IoT networks by using SAWOA as its primary tool. The delay, the amount of time, the power, the load, and the temperature were used to determine CH performance [14].

The proposed model is compared to a number of algorithms based on WOA, including CH, GSA, ABC, GA, AGSA, and PSO selection models. The simulation model's results show that SAWOA is effective in selecting a CH in order to extend network life. Beloglazov and Buyya [15] used a novel MOFGSA algorithm to choose the best CH. The energy of each node in the IoT initially supplies the packets in a significant amount for efficient routing. Fractional theory and GSA are combined in this FGSA algorithm.

The effectiveness of the algorithm is evaluated alongside that of other algorithms already in existence, such as ABC, GSA, multiparticle swarm cooperative algorithm, and so on to ensure a longer IoT node lifespan. Dhall and Agrawal [16] proposed this HEEQA algorithm to achieve a balance between system energies. Then, tune the message authentication code layer parameters to reduce system power consumption. With 190 of these IoT computers, achieving QoS is a big challenge, and maintaining power balance is critical to extending sensor life. Combining quantum PSO with modern uncontrolled gene sorting methods allows for the accomplishment of this goal. The HEEQA algorithm optimizes power consumption and improves network durability, throughput, and coverage according to simulation results. Amirtharaj et al. [17] proposed unequal clustering of time delay routing techniques to overcome WSN power consumption and data transmission problems. It's compared to others. This approach enhances network efficiency and balances energy usage, increasing network life. The proposed technique is ideal for low-latency IoT applications.

A Hy-IoT algorithm is proposed, which extends the performance of clustering to encompass actual cyber-IoT infrastructures [20]. To take advantage of the CH region, weightable election probabilities are updated based on residual strength, distance, and observed heterogeneity conditions by looking at different dynamic steps. Compared to LEACH, SEP, and Z-SEP, simulations show that Hy-IoT extends network life and improves performance. Vigorito et al. [21] proposed a new OGMAD approach that modifies the active period superframe, so that it corresponds to the data that were requested. This approach improves link utilization while guaranteeing more time slot nodes. Hu et al. [22] proposed a context knowledge-based cryptographic protocol for IoT networks that select the optimal cryptographic protocol based on data sensitivity and system requirements.

This approach reduced execution time by 68% while saving 79% on memory usage and 56% on battery usage. Hsu et al. [23] proposed an IoT- and environment-based robot architecture that allows the robot to communicate with any computer on this IoT network. Jurdak et al. [24] used the sensor routing protocol in many applications and also presented a neuro-fuzzy approach for identifying intruders in low-power WSN network. Lee and Chung [25] provided a system to detect and monitor cloud assaults that can be applied to IoT networks. Barath et al. [26] and Zhan and Lai [27] offered a unique IoT-based mechanism to improve home surveillance utilizing smartphone apps and online apps. This strategy protects consumers from home break-ins. From the above, we may conclude that despite the many CH selection models, they all spend a lot of energy. We employ a hybrid PSO-NN algorithm to optimize power consumption by selecting the optimal CH.

3. IoT-Based Adaptive Cluster Head Selection

An IoT network is composed of numerous sensor nodes, each of which has limited memory and requires a significant amount of electricity [24, 28–30]. These nodes are constantly producing data, so the battery is used more often. High-energy consumption shortens network life. One of the energy optimization strategies is choosing the optimal CH. Clustering groups sensor nodes and assigns a leader depending on criteria. A cluster is called a group and a CH is called a group leader.

In the present paper, it considers C nodes in each cluster denoted by Cj, where $J = 1, 2.., N$. Y nodes in each cluster are presented by Yj, where $J = 1, 2, \ldots, M$. TCH stands for total number of CHs. Only the TCH that has been selected is able to interact with the IoT base station. Choosing the best CH for an IoT-based WSN that would maximize uptime has increased challenging.

3.1. Mathematical Formulation of Fitness Function. Traditional WSNs like distance, delay, and power prefer CH. Load and temperature should be considered when combining WSN and IoT. CH was chosen to improve network capacity and endurance by supporting high power nodes with low load, latency, distance, and temperature. Maximizing the fitness function improves network stability and efficiency, which is given in Equation (1):

$$\begin{aligned} FFun_i = \; & wt0 * FFun_{Temp} + wt1 * FFun_{Load} + wt2 * FFun_{Ener} \\ & + wt4 * (1 - FFun_{Dist}) + wt5 * (1 - FFun_{Dela}), \end{aligned} \tag{1}$$

where the weighted parameters are wt0, wt1, wt3, wt4, and w5 and the fitness function ($FFun_i$) is the sum of those values. The following subsections illustrate the mathematically modeled calculation of the four parameters outlined during this experiment.

3.2. Energy Computation. In IoT systems, the energy that is absorbed along a single path will be split into two distinct halves at some point. The first component is the total amount of power amplifier energy, denoted by E_{am}; the other component is the total amount of energy used by the other circuit blocks, denoted by E_{cb}. The following is the formula that is used to get the total amount of energy that is consumed by each link:

$$E_c = E_{am} + E_{cb} \ldots . \tag{2}$$

It is not possible to replenish the energy contained in IoT nodes. The initial amount of energy that the IoT node has is denoted by $E_c = 0$. Every node that is part of a cluster is responsible for sending the packets to the CH. Throughout transmittal of packets from xth-specific node to yth CH as nodes lose energy. IoT nodes have receiver and transmitter hardware. A node's energy lost as a transmitter and receiver when it sends or receives data. The transmitter's energy dissipation may be from power electronics or radiophysics, while the receiver's is from radiophysics. Equation (3) defines the node as transferring X bytes of data to CH. Equation (5) demonstrates energy dissipation when CH receives X bytes from a typical node.

The resources of IoT nodes are unable to be replenished. The initial energy of the IoT node is denoted by the value $E_c = 0$. Every node that is part of the cluster is responsible for sending the packet to the CH. Each CH and node lose energy during packet transmission from the xth node to the yth CH. Each IoT node is equipped with receiver and transmitter hardware. Energy is lost as a transmitter and receiver when a node transmits or receives specific data. Energy dissipation in the transmitters could be due to power electronic equipment.

There are two distinct methods for releasing stored energy. The first scenario is depicted in Equation (3) and occurs when the node is being transferred X bytes of data to the CH. Equation (5) describes the energy dissipation that occurs when the CH gets the information of X bytes from the conventional node.

$$E_c\left(Dist_{N0}^{a0}\right) = E_{ee} * x) + E_{fes} * x)\left|\left|Dist_{N0}^{a0} - Dist_{CluH}^{n0}\right|\right|, \tag{3}$$

where $E_c\left(Dist_{N0}^{a0}\right)$ is the energy dissipation of the a0th conventional node, E_{ee} represents electronic energy, and E_{fes} represents free energy house.

and,

$$E_{ee} = E_{te} + E_{dae}, \tag{4}$$

where E_{te} is being transmitter of energy and E_{dae} is information aggregation energy.

$$E_e\left(Dist_{CluH}^{n}\right) = E_{ee} \, {}^{*}\mathrm{x}. \tag{5}$$

Following the data send and receive operations, the energy levels in all of the conventional nodes and the CH should be altered. Equation (6) gives the modified energy available inside the traditional node to the CH. Equation (7) gives the energy modifications in CH after receiving the information.

Information is sent to CH through the conventional node until the energy of the node becomes nonzero. A dead node has its energy level zero and Equation (8) presents the fitness function related to energy. If the node has high energy, CH should be chosen.

$$E_{E+1}\left(Dist_{N0}^{a0}\right) = E_E\left(Dist_{N0}^{a0}\right) - E\left(Dist_{N0}^{a0}\right). \tag{6}$$

Equation (7) defines the receiving data from the typical node, and it provides the changed energy that is readily available in CH.

$$E_{E+1}\left(Dist_{CluH}^{n0}\right) = E_E\left(Dist_{CluH}^{n0}\right) - E\left(Dist_{CluH}^{n0}\right). \tag{7}$$

The regular node will continue to transfer the data to the CH as long as the energy level of the node is greater than zero. A node is considered to be "dead" whenever its energy level reaches 0, at which point it is removed from the network. Equation (8) is used to represent the fitness function for energy.

$$FitFun_{Ener} = \frac{1}{Y}\left\{\sum_{i=1}^{Y} E_c\left(Dist_{N0}^{a0}\right)\right\} + \frac{1}{T_{CluH}}\left\{\sum_{n=1}^{CluH} E_e\left(Dist_{CluH}^{n0}\right)\right\}. \tag{8}$$

To be selected as CH, the node's energy must be high.

3.3. Compute Distance. The fitness function of the space between detector nodes and the IoT Bstn is modeled mathematically in Equation (9), which explains how it works. In order to pick the CH that is the least difficult, the distance that separates the nodes from the IoT Bstn should be the shortest possible.

$$FitFun_{Dist} = \sum_{n=1}^{CluH} \frac{\left|\left|Dist_{N0}^{a0} - Dist_{CluH}^{n0}\right|\right| + \left|\left|Dist_{CluH}^{n0} - Dist_{BStn}\right|\right|}{(P * Q)}, \tag{9}$$

where $Dist_{N0}^{a0} - Dist_{CluH}^{n0}$ is the difference between the a_0th traditional node and the n0th, $Dist_{CluH}^{n0} - Dist_{BStn}$ is distance between the nth CH and the IoT Bstn, and P and Q (in denominator) are the dimension ranges (in meters).

3.4. Delay Computation. In order to select the most straightforward CH, the delay ought to be as brief as is practicable. It is required that the delay can be between 0 and 1 ms. Because the length of the delay is proportional to the number of nodes in the cluster, you should eliminate nodes from the cluster in order to shorten the length of the delay. Equation (10) provides a mathematical description of the fitness function for delay transmission between devices connected to the IoT and CH. The dividend is a representation of the majority of the information that was transferred from CH to Bstn, while the divisors represent the individual nodes.

$$FitFun_{Dela} = \frac{Max\sum_{n=1}^{T_{CluH}} CluH_{n0}}{Z}. \tag{10}$$

4. PSO-Based Parameter Optimization Model

Kennedy and Eberhart created the PSO in 1995, and it was influenced by the actions of animal groups such as fish, swarms, and bird [23].

The PSO is an iterative optimization approach that is easy to apply, scalable, resilient, and fast to converge. It uses simple mathematical operators and is memory and performance efficient [50]. The PSO method is composed of a horde of particles, each of which stands for a different feasible approach to resolving the issue at hand.

The location, velocity, and fitness value of each particle are all determined by an optimization function. The value of the particle's velocity gives information about both the direction and the distance of its movement. The procedure described in the proposed work begins with the initialization of a group of random particles for L, N, and E. After that, it updates generations in order to search for the best possible answer. Each particle iteration uses the two "best" values. First is the best, thus, far. The particle swarm optimizer tracks each particle's best value so far. gbest is a global best value (the global best position). Each particle updates its location and velocity by monitoring pbest and gbest and the velocity and location of the particles are modified using the below mentioned equations:

$$\begin{aligned} V_{i,j}(t+1) = {} & w * V_{i,j}(t) + c1 * r1 * \left(p_{ij}^{lB}(t) - x_{i,j}(t)\right) \\ & + c2 * r2 * p_j^{gB}(t) - x_{i,j}(t)), \end{aligned} \tag{11}$$

$$x_{i,j}(t+1) = x_{i,j}(t) + v_{i,j}(t+1), \tag{12}$$

where discovery and exploitation abilities of PSOs are impartial by the inertia weight ω. Two distinct random numbers, viz., $r1$ and $r2$ are (0, 1), contributing to the algorithm's existence. The acceleration coefficients ($c1$, $c2$) force each

Initialization of x_i, v_i, c_1, c_2, l_b, u_b
While $I_{current} <= I_{\max}$ **do**
1. Map x_i into **W** and b
2. Evaluate Equations 11 and 12. This phase is called training of FFNN
3: Fitness (i.e., FFNN error or MSE) can be obtained using Equation 1.
if pBestScore > Fitness **then**
| pBestScore = Fitness and pBestPosition = x.
Else
End
If gBestScore > fitness **then**
| gBestScore = Fitness and gBestPosition = x.
Else
End
4: Now calculate w by using Equation 11.
5: Update velocity $v_{ij}(t+1)$ and position $x_{ij}(t+1)$ of particles according to Equations 11 and 12, respectively.
End
Final: PSO's best particle positions (pBest) are the ($\boldsymbol{W}$ and b) for FFNN.

Algorithm 1: Pseudocode for FFNN training using PSO.

particles in the most favorable direction. t denotes the latest version.

W * $v_{i,j}(t)$: explores PSO's new territory, $c1 * r1 * (p_{ij}^{lB}(t) - x_{i,j}(t))$ represents one's own thoughts, and $c2 * r2 * p_j^{gB}(t) - x_{i,j}(t))$ represents particle cooperation.

Algorithm 1 describes our suggested PSO-based parameter selection technique for feedforward NNs, which we have developed and the same returns a configuration that is near optimal. The algorithm is described in terms of two use case scenarios, which are referred to as smart city services and proxy IoT services, respectively. Section 5 will go into further detail on these topics.

5. Results and Discussion

CH selection in IoT may be simulated with MATLAB R2015a. The Xively IoT platform supplies the data that are used in this simulation. In the course of this research, a number of performance indicators, including energy, load, temperature, the number of active nodes, and cost functions, were taken into consideration in order to determine the best CH.

The simulation is based on a field that is 100×100 m in size. The IoT big topic is expected to be at the forefront of the study sector. In the following part of this section, a comparison is made between the results obtained by the suggested model and those obtained by previously developed algorithms such as SVM, GSA, and BPSO algorithms.

5.1. Performance Evaluation: The Number of Alive Nodes. Figure 2 shows how the number of active network nodes is used to evaluate and compare the proposed model to existing models. All nodes are valid during the first 1,000 iterations of

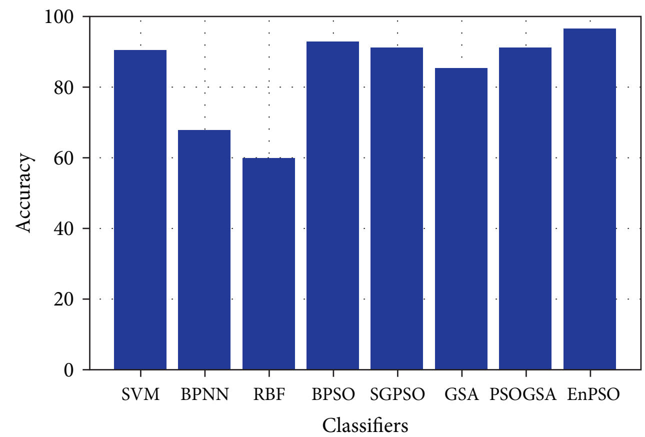

Figure 2: Evaluation of performance established on the total number of functioning nodes.

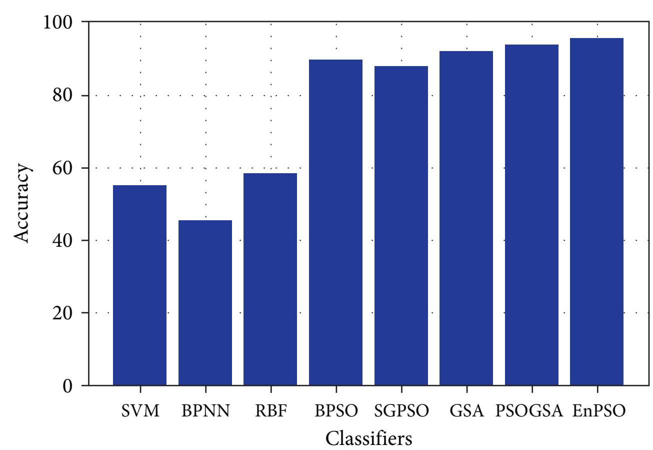

Figure 3: Performance evaluation established on load metric. The temperature produced by the sensor nodes will be lower when the load on the CH is lower.

the simulation, as shown in the following figure. For all models, the number of active nodes steadily decreased after 1,000 iterations. The current model has no live nodes after 1,700 iterations. Even after 1,700 iterations, the model that was proposed still contained approximately 20 nodes that were active. The suggested model keeps more nodes alive before the final iteration, extending network longevity.

5.2. Performance Evaluation: The Load Metric. The results of the suggested model with CHs functioning as a load may be shown in Figure 3. If the load is distributed uniformly over all of the CHs, then the efficiency of the IoT network will be at its highest possible level. According to this figure, the load is distributed very evenly over all CHs using the methodology that was provided. In addition to this, the proposed model optimizes the load in each iteration of the process. Even after 1,500 iterations, the CH load is still significantly lower than the version that is currently in use.

The IoT network consumes less energy as a result of this, which increases the network's efficiency.

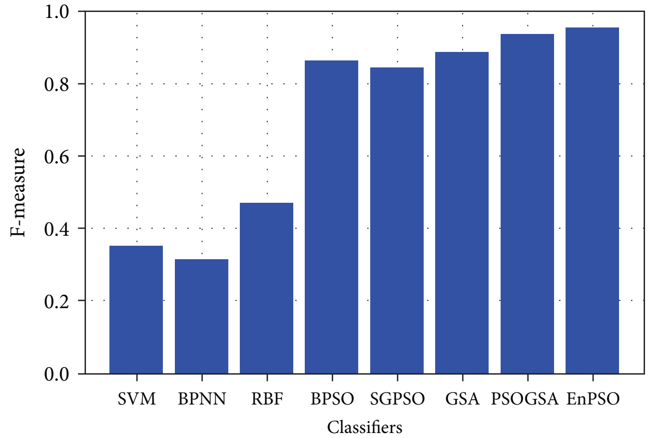

FIGURE 4: Performance evaluation established on energy metric.

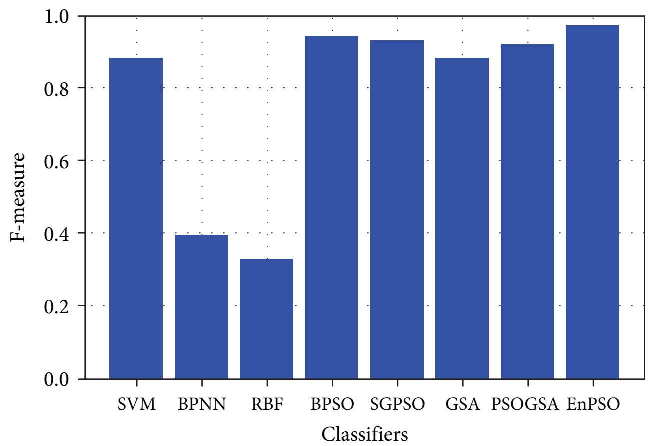

FIGURE 5: Performance evaluation based on cost function.

5.3. Performance Evaluation: The Energy Metric. The performance of the suggested energy-based model is shown in Figure 4 to the performance of other models that already exist. At the beginning, each model assumes that the energy of the network is 0.5 J. The energy required to complete a task decreases proportionately with the number of iterations performed. In comparison to the model that is currently being used, the proposed model consistently results in a greater amount of energy being present in the network after each iteration. This helps ensure that the network will continue to function effectively in the years to come.

5.4. Performance Evaluation: The Cost Function. Figure 5 depicts the cost function-based output of the proposed model. The convergence of an algorithm is determined by the cost function.

As the number of iterations in an algorithm rises, the convergence of the method should generally get better.

The proposed model outperforms the current model in terms of convergence, as can be seen in the graph. As shown in Figures 2–5, the proposed model outperforms the current model for all the measures considered. Existing models make use of "blind operators" for the sake of manipulation, which are independent of the fitness function. Simulated annealing, which plays the role of an operator in the PSO-NN process, makes it feasible to guarantee that the blind operator is replaced with a local search that uses the solution as the initial state. This makes it possible to guarantee that the blind operator is replaced. After completing the primary objective, the enhanced solution is implemented in place of the initial one. The utilization of the PSO algorithm is consequently improved as a consequence of the process of simulated annealing. As a consequence of this, the simulated annealing algorithm contributes to an improvement in the effectiveness of PSO in locating the best solution. The end conclusion is that the suggested model beats existing approaches when it comes to optimizing the effectiveness of IoT networks.

5.5. Comparison with the State-of-the Art Approaches. Here are some comparisons of hybrid PSO with other approaches in terms of IoT-based smart cities:

(1) Genetic algorithms: Both PSO and genetic algorithms are metaheuristic optimization algorithms that can be used to optimize IoT-based smart cities. However, hybrid PSO combines PSO with genetic algorithms to improve the performance of the algorithm. Hybrid PSO can be more efficient than genetic algorithms alone because it uses PSO to explore the search space and genetic algorithms to exploit the best solutions.

(2) Ant colony optimization: It is another optimization technique that can be used to optimize IoT-based smart cities. Like genetic algorithms, hybrid PSO combines PSO with ant colony optimization to improve the performance of the algorithm. Hybrid PSO can be more efficient than ant colony optimization alone because it uses PSO to explore the search space and ant colony optimization to exploit the best solutions.

(3) Convex optimization: It is a mathematical optimization technique that can be used to optimize IoT-based smart cities. However, convex optimization requires that the objective function and the constraints be convex, which may not always be the case in real-world scenarios. Hybrid PSO does not have this limitation and can be used to optimize a wide range of objective functions and constraints.

(4) Reinforcement learning: It is a machine learning technique that can be used to optimize IoT-based smart cities. However, reinforcement learning requires a large amount of data to train the model, which may not always be available in real-world scenarios. Hybrid PSO does not require training data and can be used to optimize IoT-based smart cities in real time.

In conclusion, hybrid PSO can be a powerful optimization technique for IoT-based smart cities. It combines the strengths of PSO with other optimization techniques to improve the performance of the algorithm and can be used to optimize a wide range of tasks in real time. However, the choice of optimization technique ultimately depends on the specific requirements of the problem at hand.

6. Conclusion and Future Direction

In spite of the fact that the IoT has enormous promise in a large number of different applications in the modern world, there are a lot of hurdles to jump over first. To make the IoT more robust, there are a number of difficulties that need to be resolved, including those pertaining to data access, hardware compatibility, and optimization of power consumption. For the purpose of this investigation, we settled on concentrating on the challenge of energy optimization. In order to solve this problem, this research paper employs a hybrid metaheuristic algorithm known as PSO-NN to optimize the sensor power consumption of this IoT-based WSN. For the purpose of modeling this IoT network, this research makes use of the Xively IoT platform. There have been a total of 2,000 iterations of the IoT. In this piece of work, numerous performance parameters, like cost function, residual energy, number of active nodes, temperature, and load, are used to determine the best CH for the operation of IoT network 370. It then compares the proposed method with various existing methods. The results of the experiments demonstrate that the proposed method is superior to the method that is currently being used. In the future, determining the best CH may also involve taking into account a number of other performance parameters, including link lifetime, node density, and latency. When we utilize this strategy in real-time applications, such as those employed in a variety of sensors, we are also able to test the scalability of the work that we have planned.

Future research work will address the power optimization for each individual clustering IoT networks, and related security issues.

Abbreviations

CH:	Cluster head
PSO:	Particle swarm optimization
ANN:	Artificial neural network
QoS:	Quality of service
WSN:	Wireless sensor network
BT:	Base station
PSO-NN:	Particle swarm optimization-neural network.

Data Availability

Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] F. Cicirelli, A. Guerrieri, G. Spezzano, and A. Vinci, "An edge-based platform for dynamic smart city applications," *Future Generation Computer Systems*, vol. 76, pp. 106–118, 2017.

[2] B. N. Silva, M. Khan, and K. Han, "Towards sustainable smart cities: a review of trends, architectures, components, and open challenges in smart cities," *Sustainable Cities and Society*, vol. 38, pp. 697–713, 2018.

[3] M. I Ali, "Multi-layer cross domain reasoning over distributed autonomous IoT applications," *Open Journal of Inter390 Net Of Things (OJIOT)*, vol. 3, no. 1, pp. 75–90, 2017.

[4] S. Soursos, I. P. Zarko, P. Zwickl, I. Gojmerac, G. Bianchi, and G. Carrozzo, "Towards the cross-domain interoperability of IoT platforms," in *2016 European Conference on Networks and Communications (EuCNC)*, pp. 398–402, IEEE, Athens, 2016.

[5] Y. Liu, C. Yang, L. Jiang, S. Xie, and Y. Zhang, "Intelligent edge 400 computing for IoT-based energy management in smart cities," *IEEE Network*, vol. 33, no. 2, pp. 111–117, 2019.

[6] D. S. Roy, R. K. Behera, K. H. K. Reddy, and R. Buyya, "A context aware fog enabled scheme for real-time cross-vertical IoT applications," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2400–2412, 2018.

[7] A. Bomnale and S. Malgaonkar, "Power optimization in wireless sensor networks," in *2018 International Conference on Communication Information and Computing Technology (ICCICT)*, pp. 1–6, IEEE, New York, 2018.

[8] J. A. Khan, H. K. Qureshi, and A. Iqbal, "Energy management in wireless sensor networks: a survey," *Computers & Electrical Engineering*, vol. 41, pp. 159–176, 2015.

[9] N. Wang, B. Varghese, M. Matthaiou, and D. S. Nikolopoulos, "ENORM: a framework for edge node resource management," *IEEE Transactions on Services Computing*, vol. 13, no. 6, pp. 1086–1099, 2020.

[10] T. H. Luan, L. Gao, Z. Li, Y. Xiang, G. Wei, and L. Sun, "Fog computing: focusing on mobile users at the edge," 2015.

[11] I. Bouzguenda, C. Alalouch, and N. Fava, "Towards smart sustainable cities: a review of the role digital citizen participation could play in advancing social sustainability," *Sustainable Cities and Society*, vol. 50, Article ID 101627, 2019.

[12] S. E. Bibri, "The IoT for smart sustainable cities of the future: an analytical framework for sensor-based big data applications for environmental sustainability," *Sustainable Cities and Society*, vol. 38, pp. 230–253, 2018.

[13] V. Dinesh Reddy, G. R. Gangadharan, and G. S. V. R. K. Rao, "Energy-aware virtual machine allocation and selection in cloud data centers," *Soft Computing*, vol. 23, no. 6, pp. 1917–1932, 2019.

[14] G. Cao, "Topology-aware multi-objective virtual machine dynamic consolidation for cloud datacentre," *Sustainable Computing: Informatics and Systems*, vol. 21, pp. 179–188, 2019.

[15] A. Beloglazov and R. Buyya, "Energy efficient allocation of virtual machines in cloud data centers," in *2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing*, pp. 577-578, IEEE, 2010, May.

[16] R. Dhall and H. Agrawal, "An improved energy efficient duty cycling algorithm for IoT based precision agriculture," *Procedia Computer Science*, vol. 141, pp. 135–142, 2018.

[17] I. Amirtharaj, T. Groot, and B. Dezfouli, "Profiling and improving the duty-cycling performance of Linux-based IoT devices," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–29, 2019.

[18] B. Afzal, S. A. Alvi, G. A. Shah, and W. Mahmood, "Energy efficient context aware traffic scheduling for IoT applications," *Ad Hoc Networks*, vol. 62, no. 62, pp. 101–115, 2017.

[19] A. Kozłowski and J. Sosnowski, "Energy efficiency trade-off between duty-cycling and wake-up radio techniques in IoT networks," *Wireless Personal Communications*, vol. 107, no. 4, pp. 1951–1971, 2019.

[20] C. Moser, L. Thiele, D. Brunelli, and L. Benini, "Adaptive power management in energy harvesting systems," in *2007*

Design, Automation Test in Europe Conference Exhibition, pp. 1–6, IEEE, 2007.

[21] C. M. Vigorito, D. Ganesan, and A. G. Barto, "Adaptive control of duty cycling in energy-harvesting wireless sensor networks," in *2007 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, pp. 21–30, IEEE, 2007.

[22] J. Hu, Q. Qian, A. Fang, S. Wu, and Y. Xie, "Optimal data transmission strategy for healthcare-based wireless sensor networks: A stochastic differential game approach," *Wireless Personal Communications*, vol. 89, no. 4, pp. 1295–1313, 2016.

[23] J. Hsu, S. Zahedi, A. Kansal, M. Srivastava, and V. Raghunathan, "Adaptive duty cycling for energy harvesting systems," in *Proceedings of the 2006 international symposium on Low power electronics and design*, pp. 180–185, IEEE, 2006.

[24] R. Jurdak, P. Corke, D. Dharman, and G. Salagnac, "Adaptive GPS duty cycling and radio ranging for energy-efficient localization," in *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, pp. 57–70, 2010.

[25] D. Lee and K. Chung, "Adaptive duty-cycle based congestion control for home automation networks," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 1, pp. 42–47, 2010.

[26] S. R. Barath, B. U. Usha, G. B. Arpitha, V. Kumar, and S. M. Yaseen, "Strategies for Congestion Control in Wireless Sensor Networks: A Survey," in *2019 International Conference on Smart Systems and Inventive Technology (ICSSIT)*, pp. 629–633, IEEE, 2019.

[27] C. Zhan and H. Lai, "Energy minimization in Internet-of-Things system based on rotary-wing UAV," *IEEE Wireless Communications Letters*, vol. 8, no. 5, pp. 1341–1344, 2019.

[28] S. Sendra Compte, J. L loret, M. Garcıa Pineda, and J. F. Toledo Alarcon, "Power saving and energy optimization techniques for wireless sensor networks," *Journal of Communications*, vol. 6, no. 6, pp. 439–459, 2011.

[29] S. S. Beheshtiha, H. P. Tan, and M. Sabaei, "Opportunistic routing with adaptive harvesting-aware duty cycling in energy harvesting WSN," in *The 15thInternational Symposium on Wireless Personal Multimedia Communications*, pp. 90–94, IEEE, 2012.

[30] O. Briante, A. M. Mandalari, A. Molinaro, G. Ruggeri, J. AlonsoZarate, and F. Vazquez-Gallego, "Duty-cycle optimization for machine-to-machine area networks based on frame slotted-ALOHA with energy harvesting capabilities," in *European Wireless 2014; 20th European Wireless Conference*, pp. 1–6, IEEE, 2014.

[31] C. H. S. Oliveira and M. D. Castro, "Autonomic Duty Cycling for Target Tracking in a Bio-Inspired Wireless Sensor Network," in *International Conference on Systems (ICONS'15)*, pp. 112–118, ICONS.

[32] C. H. Oliveira, Y. Ghamri-Doudane, and S. Lohier, "A duty cycle self-adaptation algorithm for the 802.15. 4 wireless sensor networks," in *Global Information Infrastructure Symposium-GIIS. 2013*, pp. 1–7, IEEE, 2013, October.

[33] J. Tang, H. Tang, X. Zhang et al., "Energy minimization in D2D-Assisted Cache500 enabled internet of things: a deep reinforcement learning approach," *IEEE Transactions on Industrial Informatics*, 2019.

[34] S. Vergis, V. Komianos, G. Tsoumanis, A. Tsipis, and K. Oikonomou, "A low-cost vehicular traffic monitoring system using fog computing," *Smart Cities*, vol. 3, no. 1, pp. 138–156, 2020.

[35] K. Ma, A. Bagula, C. Nyirenda, and O. Ajayi, "An IoT-based fog com510 putting model," *Sensors*, vol. 19, no. 12, Article ID 2783, 2019.

[36] O. Skarlat, M. Nardelli, S. Schulte, and S. Dustdar, "Towards qos-aware fog service placement," in *2017 IEEE 1st international conference on Fog and Edge Computing (ICFEC)*, pp. 89–96, IEEE, 2017.

[37] V. B. Souza, X. Masip-Bruin, E. Marín-Tordera et al., "Towards a proper service placement in combined Fog-to-Cloud (F2C) architectures," *Future Generation Computer Systems*, vol. 87, pp. 1–15, 2018.

[38] Q. T. Minh, D. T. Nguyen, A. Van Le, H. D. Nguyen, and A. Truong, "Toward service placement on Fog computing landscape," in *2017 4th NAFOSTED conference on information and computer science*, pp. 291–296, IEEE, 2017.

[39] M. I. Naas, P. R. Parvedy, J. Boukhobza, and L. Lemarchand, "iFogStoran IoT data placement strategy for fog infrastructure," in *2017 IEEE 1st International Conference on Fog and Edge Computing (ICFEC)*, pp. 97–104, IEEE, 2017.

[40] M. Taneja and A. Davy, "Resource aware placement of IoT application modules in Fog-Cloud Computing Paradigm," in *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, pp. 1222–1228, IEEE, 2017.

[41] I. Lera, C. Guerrero, and C. Juiz, "Availability-aware service placement policy in fog computing based on graph partitions," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 3641–3651, 2019.

[42] K. A. Eldrandaly, M. Abdel-Basset, and L. Abdel-Fatah, "PTZ-Surveillance coverage based on artificial intelligence for smart cities," *International Journal of Information Management*, vol. 49, pp. 520–532, 2019.

[43] R. K. Behera, K. H. K. Reddy, and D. S. Roy, "A novel context migration model for fog-enabled cross-vertical IoT applications," in *International Conference on Innovative Computing and Communications*, pp. 287–295, Springer, Singapore, 2020.

[44] K. H. K. Reddy, R. K. Behera, A. Chakrabarty, and D. S. Roy, "A service delay minimization scheme for QoS-constrained, context-aware unified IoT applications," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 10527–10534, 2020.

[45] O. Skarlat, M. Nardelli, S. Schulte, M. Borkowski, and P. Leitner, "Optimized IoT service placement in the fog," *Service Oriented Computing and Applications*, vol. 11, no. 4, pp. 427–443, 2017.

[46] M. Zubair, N. Javaid, M. Ismail, M. Zakria, M. A. Zaheer, and F. Saeed, "Integration of cloud-fog based platform for load balancing using hybrid genetic algorithm using bin packing technique," in *International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*, pp. 279–292, Springer, 2018.

[47] Y. Sun, F. Lin, and H. Xu, "Multi-objective optimization of resource scheduling in fog computing using an improved NSGA-II," *Wireless Personal Communications*, vol. 102, no. 2, pp. 1369–1385, 2018.

[48] C. Guerrero, I. Lera, and C. Juiz, "Evaluation and efficiency comparison of evolutionary algorithms for service placement optimization in fog architectures," *Future Generation Computer Systems*, vol. 97, pp. 131–144, 2019.

[49] A. Kaur and S. K. Sood, "Cloud-Fog based framework for drought prediction and forecasting using artificial neural network and genetic algorithm," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 32, no. 2, pp. 273–289, 2020.

[50] A. Gyrard, S. K. Datta, C. Bonnet, and K. Boudaoud, "Cross domain Internet of Things application development: M3 framework and evaluation," in *2015 3rd International Conference on Future Internet of Things and Cloud*, pp. 9–16, IEEE, 2015.