scientific reports

Check for updates

OPEN

# Implementing a smart city using internet of things based on wildwood combined with fractional order-based golden search algorithm

Tianlan Wang[1] & Zhiwen Zhao[2]✉

**The fast expansion of Internet of Things (IoT) devices in urban environments has resulted in a dramatic increase in both the volume and complexity of data produced, necessitating the implementation of sophisticated data analytics and machine learning methodologies to fully realize the advantages of smart cities. The incorporation of IoT sensors and devices has facilitated the establishment of extensive, dynamic, and diverse networks, which present considerable challenges for data analysis and decision-making processes. In response to these challenges, machine learning algorithms have surfaced as a feasible solution, capable of discerning intricate patterns and relationships within the data. Nonetheless, the efficacy of these algorithms is significantly influenced by the precise tuning of hyperparameters, a task that can be quite complicated, particularly within IoT-enabled smart cities. This study concentrates on the creation of an innovative optimization framework that integrates the WildWood algorithm with a fractional-order variant of the Golden Search Algorithm for hyperparameter optimization. The proposed framework is measured through simulations in a smart city traffic management background, resulting in notable reductions in latency (up to 30%), energy consumption (up to 25%), and enhancements in throughput (up to 20%) when compared to conventional optimization techniques. Moreover, the optimized WildWood model achieves a Mean Squared Error (MSE) of 0.85, indicating its proficiency in accurately forecasting traffic flow patterns. The results underscore the effectiveness of the proposed framework in enhancing the reliability, efficiency, and sustainability of IoT-enabled smart city systems.**

**Keywords** Smart city, Internet of Things (IoT), Machine learning, WildWood algorithm, Fractional order golden search algorithm, Optimization, Data analysis, Simulation

The rapid urbanization has formed an increasing demand for effective and sustainable city management. Smart cities, which are equipped with interconnected devices known as the Internet of Things (IoT), offer a promising solution to address these challenges[1]. These interconnected sensors gather a large amount of data that provides valuable insights into various aspects of city life, such as traffic flow, energy usage, and environmental conditions[2]. However, effectively utilizing this data requires strong techniques for data analysis and model optimization.

Smart cities data has the potential to revolutionize city living. Picture traffic signals adjusting in real-time to alleviate congestion, or energy consumption being optimized based on grid demands and weather, leading to sustainability and cost savings[3]. By harnessing this data, city officials can make informed decisions for efficient resource allocation, proactive issue resolution, and enhanced citizen services. However, analyzing this data comes with its own set of challenges. The sheer volume, diversity, and real-time nature of the data can overwhelm traditional methods. Additionally, the inherent noise and non-linear relationships in sensor data call for robust techniques. To fully unlock the potential of this data, we must optimize the analytical models used[4]. These models depend on hyperparameters that significantly impact their performance[1]. Traditional data analysis methods may struggle with the complexity and volume of data generated by smart city IoT networks.

[1]Hangzhou Polytechnic, Hangzhou 311402, Zhejiang, China. [2]Zhejing guangsha vocational and technical university of construction, Dongyang, Zhejiang 322100, China. ✉email: hnhyzzw@163.com

Furthermore, obtaining accurate and reliable results depends on optimizing the performance of the chosen analytical models.

Alqahtani et al.[5] conducted a research with the purpose of developing Urban Waste System of Management based on Internet of Things. IoT devices were utilized to oversee activities of human beings and aided in management of waste. Data related to a city has been gathered and analyzed using a long short-term recurrent neural network optimized by cuckoo search. The network enabled the examination of waste category, truck dimensions, and waste origin, facilitating the alerting of management centers of waste to take necessary actions. The IoT-based waste management procedure's effectiveness has been assessed via an experimental study. It was determined that the system prioritized processing the bins with minimal error value of 0.16 and the highest accuracy value of 98.4% in as little as 15 min.

Annadurai et al.[6] suggested a kind of method for transmitting secure data and diagnosing an invader within a biometric system of authentication through extracting features employing categorization. The biometric data has been processed to eliminate noise, normalization, and smoothening. The processed features of data have been derived employing Kernel-based Principal Component Analysis (KPCA). Following this, the processed attributes were categorized employing the convolutional VGG-16 Net framework. The whole model was secures employing a Deterministic Trust Transfer Protocol (DTTP). The findings of the study indicated that the suggested approach yielded improved intrusion diagnosis outcomes. The study could achieve the values of 96%, 85%, 92%, 80%, and 46% for accuracy, f-score, Precision, recall, and an RMSE.

In an attempt, Chen et al.[7] endeavored to distribute learning structure by employing edge intelligence and enhancing networking capability of intelligent terminal nodes that were organized on some computers, which were ordinary rather than specific hardware, like GPUs. The suggested method, which utilized intelligent edge techniques and multi-core CPU, reduced training costs and time while effectively utilizing resources of computer. Additionally, the most efficient method took into account the distributed communication system of edge computation and optimized the topology of network by employing the contributions of smart terminal nodes. It was demonstrated that the approach exceled across various topologies and surpassed other advanced optimizers in enhancing the strength of IoT topologies within smart cities.

Li et al.[8] aimed to carry out the BDA (Big Data Analysis) on the huge information that were produced within the smart city Internet of Things (IoT), protect processing of data, and cause the smart city alter good supremacy's direction. It was revealed from the findings that the efficacy of energy rose and decreased when the lowest energy $\alpha_0$ rose. The forecasting accuracy of the network was scrutinized. Then, it was illustrated that the accuracy value of the recommended model could reach 97.80%.

Cepeda-Pacheco and Domingo[9] suggested a model based on deep learning. The suggested multi-label deep learning categorizer performed better than other networks, including extra tree, decision tree, random forest, and k-nearest neighbor. The model could gain the values of 99.7%, 0.5%, 99.9%, 99.9%, and 99.8% for accuracy, loss, recall, precision, and F1-score for case (a). Then, the model could suggest the values of 99.5%, 3.7%, 99.7%, 99.8%, and 99.8 for accuracy, loss, recall, F1-score, and precision for case (b).

Despite the notable progress in machine learning for smart city applications, a key issue persists: enhancing the performance of these models. The success of a machine learning model greatly relies on its hyperparameters-the configurations that govern the learning process. Although there are different optimization techniques available, there is an ongoing endeavor to discover more effective and reliable methods. Conventional optimization strategies may encounter challenges such as being trapped in local optima (less than optimal solutions) and failing to attain the model's highest potential performance.

This research investigates the use of machine learning, specifically the WildWood algorithm, for analyzing data collected from diverse IoT sensors within a smart city environment. Machine learning algorithms have great potential for extracting meaningful patterns and relationships from complex datasets. WildWood, in particular, is skilled at handling data with inherent noise or non-linearities, which are often present in real-world sensor data. However, the performance of any machine learning model heavily relies on its hyperparameters, which are the settings that control the learning process of the model. Optimizing these hyperparameters is essential to ensure the model achieves optimal performance on the target data.

This study introduces an innovative method that utilizes the WildWood algorithm for data analysis and presents a Fractional Order Golden Search Algorithm for hyperparameter optimization in the smart city domain. The Fractional Order Golden Search Algorithm brings numerous benefits compared to conventional optimization techniques. It has the potential to prevent being trapped in local optima and could result in quicker convergence, thereby enhancing the efficiency of the optimization process.

This study seeks to advance smart city data analysis by merging WildWood with a robust optimization method to:

- Improving the effectiveness and precision of machine learning models used in smart city applications.
- Showing the efficiency of the Fractional Order Golden Search Algorithm for hyperparameter optimization within this field.
- Offering a versatile framework for smart city data analysis tasks, enabling data-informed decision-making to enhance city management.
- Providing superior performance relative to conventional optimization algorithms in the enhancement of traffic flow and the alleviation of congestion within smart city infrastructures.
- Managing intricate and diverse IoT data, addressing a critical challenge faced in smart city initiatives. Capabilities for real-time optimization and decision-making, thereby enhancing the efficiency and effectiveness of smart city operations.

## Methodology
The Internet of Things (IoT) is transforming the daily lives and work routines, with a significant impact on the evolution of smart cities. This section explores the fusion of IoT and optimization technology as the cornerstone of smart city deployment, with the goal of enhancing connectivity, data sharing, and overall urban governance. As depicted in Fig. 1, the approach involves a thorough examination of how IoT functions to establish a network of interconnected devices, sensors, and systems that enhance various aspects of urban living.

The fundamental concept driving this integration is to harness the capabilities of IoT devices and sensors to gather and transmit real-time data. Through the consolidation of these technologies, smart cities can capitalize on improved data exchange, efficient resource utilization, and enhanced decision-making abilities.

A key benefit of this strategy is the capacity to streamline city operations and services. For instance, IoT sensors can be utilized to monitor traffic patterns, parking availability, air quality, and energy consumption. By collecting and analyzing this information, city administrators can make informed decisions to optimize traffic flow, reduce environmental pollution, and enhance energy efficiency. Moreover, IoT-enabled smart devices and systems can furnish residents with up-to-date information and interactive services, enriching their overall quality of life.

Furthermore, the utilization of IoT facilitates the creation of intelligent systems capable of learning and adapting to changing circumstances. Through the application of advanced analytics and machine learning algorithms, smart cities can identify trends, forecast demand, and proactively address potential challenges. This not only enhances the effectiveness of city operations but also fortifies the resilience of urban infrastructure, ensuring a sustainable and future-ready urban landscape.

## Data sample
This research utilizes a dataset obtained from a wide array of Internet of Things (IoT) devices in the United Kingdom and the United States. The section offers an overview of the sample data and its main characteristics.

The sample data consists of data flows from 80 distinct IoT devices, including smart hubs, cameras, home automation systems, televisions, music devices, and various home appliances. These devices were carefully chosen to represent a broad range of functions and usage patterns commonly found in smart homes and cities[10]. By gathering data from a diverse set of devices, we aim to capture a comprehensive view of IoT behavior and interactions.

The dataset contains a total of 40,588,450 tagged instances, providing a significant amount of data for analysis. These instances were collected from 68 actively used and monitored IoT devices during the data collection period. Each instance represents a specific event, action, or interaction within the ecosystem of connected devices.

The data covers various information, including device activity, sensor readings, user interactions, and network communications. By analyzing these data flows, we can gain insights into the behavior, performance, and interconnections within the IoT environment. This includes understanding usage patterns, device interactions, and potential security or privacy concerns.

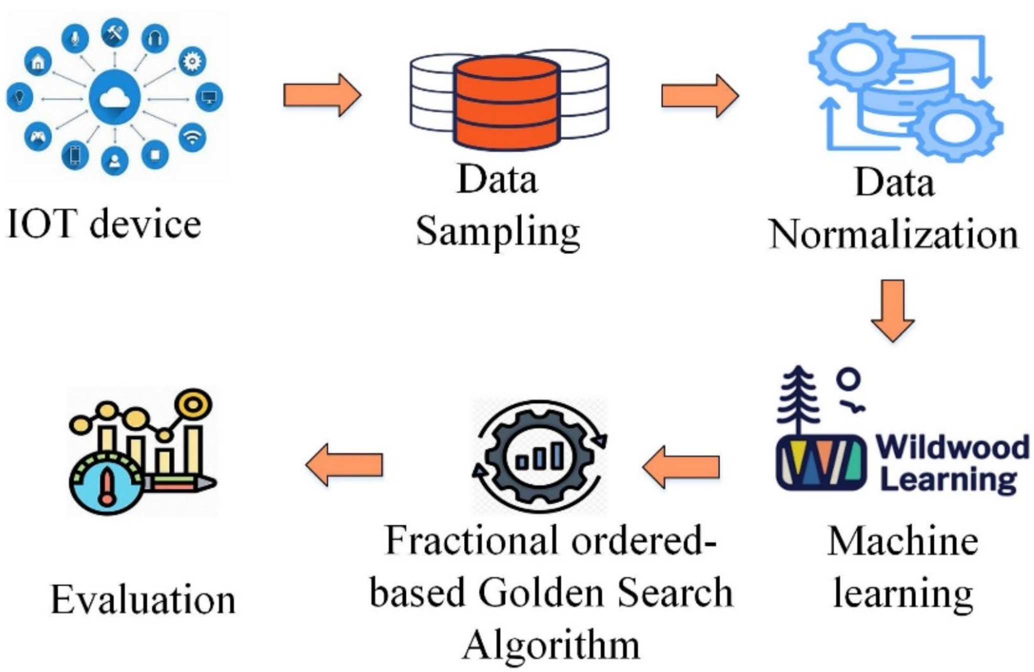


**Fig. 1**. Graphical abstract of the methodology.

It is important to note that this sample data was obtained from a separate reference, ensuring the confidentiality and integrity of the original data source. By utilizing an existing dataset, we can build upon previous research and contribute additional insights and analysis[11].

By using this sample data, our research aims to explore the potential of IoT integration in smart cities. The data provides a real-world perspective on device usage, allowing us to identify patterns, challenges, and opportunities for optimization. Through detailed analysis, we can make informed recommendations for improving the implementation and management of IoT-enabled smart cities, ultimately enhancing the efficiency and quality of urban life.

## Data normalization

In order to maintain consistency and simplify analysis, it is crucial to standardize the sample data collected from various IoT devices. Normalizing data involves adjusting and scaling it to a common range or distribution, enabling accurate comparisons and insightful data interpretation across different devices and measurement scales[12]. For our sample data, normalization methods were implemented to account for potential discrepancies in device usage, sensor accuracy, and data gathering procedures. The following steps were followed to normalize the data:

### Min–max scaling

This technique scales the data to the range between 0 and 1. Each data point is transformed using the following formula[13]:

$$Normalized\ Value = \frac{(Original\ Value - Minimum\ Value)}{(Maximum\ Value - Minimum\ Value)} \tag{1}$$

With this scaling method, we can be certain that all data is in a standardized range, enabling comparison across different devices and parameters.

### Standardization

Standardization is the process of adjusting data to have a mean of zero and a standard deviation of one. This method is especially helpful for data that conforms to a normal distribution. The formula for standardization is[14]:

$$Standardized\ Value = \frac{(Original\ Value - Mean)}{Standard\ Deviation} \tag{2}$$

Standardization is useful for pinpointing outliers and gaining insight into the varying significance of different variables.

### Logarithmic transformation

When dealing with data that has a skewed distribution, applying a logarithmic transformation can help bring it closer to a normal distribution. This transformation is particularly beneficial for addressing right-skewed data, which is frequently observed in various natural phenomena and sensor readings. The formula for the log transformation is[15]:

$$Log - transformed\ Value = log\,(Original\ Value) \tag{3}$$

Utilizing the natural logarithm results in a more balanced dataset, making it simpler to analyze with common statistical techniques.

### Categorical data encoding

Categorical variables, like device types or locations, were encoded using either one-hot encoding or label encoding methods. One-hot encoding generates binary columns for every category, showing whether a specific category is present or not. On the other hand, label encoding assigns a distinct numerical value to each category, enabling numerical analysis.

By utilizing these normalization methods, the sample data is enhanced to be more uniform and easily comparable. This process aids in recognizing trends, relationships, and irregularities within the data, resulting in more precise analysis and modeling[16]. Additionally, it simplifies the utilization of machine learning algorithms, as they tend to yield superior results with normalized data inputs. Through adherence to these procedures, we guarantee that the data is suitably adjusted, standardized, and converted, thereby enabling a thorough and impartial examination of IoT device behavior in the realm of smart city deployments.

## WildWood implementation in smart cities

In the quest to develop intelligent and efficient smart cities, the integration of the IoT is crucial. IoT allows seamless communication and interconnectivity between devices and systems, flooring the way for enhanced data exchange and automation. However, advanced data processing and decision-making capabilities are essential to fully unlock the potential of IoT. The WildWood is an advanced algorithm specifically designed to challenge the challenges and complexities inherent in IoT data analysis. It provides a robust and adaptable framework for processing and interpreting large volumes of data generated by IoT devices in smart cities. By using WildWood, valuable insights can be extracted, city operations can be optimized, and overall urban management can be enhanced.

One of the main advantages of WildWood is its ability to handle high-dimensional and heterogeneous data. IoT devices in smart cities collect and transmit data from various sensors, cameras, and other smart components, resulting in a diverse and complex dataset. WildWood excels in processing and analyzing this multi-faceted data by identifying patterns, correlations, and trends that may otherwise be difficult to discern.

Furthermore, WildWood brings a level of flexibility and context-awareness to IoT data analysis. It has the ability to learn and adapt to changes in environments and conditions within the smart city ecosystem. By incorporating feedback loops and adaptive mechanisms, WildWood can modify its decision-making processes and enhance its accuracy over time. This dynamic nature makes it well-suited for the constantly changing landscape of smart cities.

Also, WildWood provides improved accuracy and precision compared to traditional machine learning algorithms. Its unique structure and optimization techniques allow it to handle noisy and incomplete data, which are common challenges in IoT environments. Through the use of advanced feature selection and weighting methods, WildWood can identify the most relevant and informative features within the data, resulting in more accurate predictions and decisions. The integration of WildWood in smart cities can have wide-ranging benefits.

The WildWood algorithm has the capability to be mathematically structured in order to enhance IoT-driven smart city deployments. The WildWood algorithm can be formulated as the following optimization problem[17]:

$$J\left(W\right)=\frac{1}{N}\sum_{i=1}^{N}\left(Y_i-f\left(X_i,W\right)\right)^2 \tag{4}$$

$J(W)$ is a measure of the difference between the predicted values and the actual values, $N$ specifies the total number of IoT devices, $Y_i$ determines the target variable (e.g. optimized traffic flow, energy consumption, or public safety metric), $X_i$ represents the feature value of the $ith$ IoT device, $f$ signifies the WildWood model mapping input data to the target variable, $W$ defines the set of weights that assigned to each feature to show their relative importance, The weights are used to combine the features of the input data $X_i$ to make predictions, and the hyperparameters for the $j\left(w\right)$ for optimal selection are: the number of trees (T), the Maximum Tree Depth (D), the Minimum Samples per Leaf (L), the Learning Rate (η), the Regularization Strength (λ).

The goal of the function is to minimize the sum of squared differences between the target variable $Y_i$ and the predicted values from the WildWood model $f$. This will help us find the best weights $W$ and model parameters to reduce the overall error.

In this study, a fractional Order-based variant of the Golden Search Algorithm has been used for this purpose.

The WildWood model is designed to handle complex and heterogeneous data to predict the optimal values of the hyperparameters that minimize the error function, i.e., J(W). This measures the difference between the predicted values and the actual values. The purpose of using the Fractional Order-based Golden Search Algorithm is to find the set of hyperparameters $W$ to minimize the error function $J(W)$.

To achieve this, the WildWood model predicts the values of the hyperparameters $W$ that correspond to the minimum error, taking the feature vector $X_i$ as input and outputting a predicted value of the hyperparameters $W$. The predicted values of the hyperparameters $W$ are then used to compute the error function $J(W)$.

The Fractional Order-based Golden Search Algorithm then uses the computed error function to update the hyperparameters $W$, iteratively updating them until the error function is minimized. The use of the WildWood model in the Fractional Order-based Golden Search Algorithm helps to reduce the error in several ways, including improved prediction accuracy, reduced overfitting, and efficient exploration of the search space, ultimately leading to a more accurate and efficient hyperparameter optimization process.

## Fractional ordered-based golden search algorithm

### Background

Myriad random optimization techniques have been created to find optimal problem solver that usually include generating a group of logical solution, primary population, by employing a stochastic procedure. The fitness value of the present problem solvers has been evaluated within all iterations; in addition, they have been enhanced by the use of numerous formulas that make up main elements of the optimization technique. The procedure of iterative enhancement keeps on by the time a satisfactory criterion of termination has been met.

In the present research, the Golden Search Optimizer (GSO), a novel optimizer on the basis of bio-inspired algorithms, has been offered. The present algorithm integrates the benefits of earlier techniques, such as Sine Cosine Algorithms (SCA) and Particle Swarm Optimization (PSO), to strike a balance between global search and local search, which prevents from premature convergence.

The approach enhanced positions of objects by employing step size variables just like velocity within PSO. On the other hand, SCA have been employed in the place of stochastic values. Oscillation properties of the present functions permit an item to go toward another once and effectively employ the region between 2 problem solvers.

Moreover, global search abilities are probably developed through expanding the magnitude of the present functions to alter the situation of a solution that is exterior to the determined region. The GSO method has a scheme that is easy to conduct and outperforms previous bio-inspired methods regarding reaching the optimal global solution. It starts with a stochastic solution and updates the state of all individuals after each iteration, meeting the termination criterion with the help of a step size restriction. The GSO has been found to be a global optimizer that tries to generate an optimum solution by striking equilibrium between local search and global search. The approach is composed of 3 main components, including candidate renewal, candidate initialization, and candidate assessment.

- Phase 1: Initialization of candidates

The search process of the GSO commences with creating a group of individual within solution space by employing a determined optimizer. These individuals have been selected in a random manner to commence the procedure of search that guarantees several probable outcomes for the issue of optimization. The individuals are simulated in the following manner[18]:

$$\begin{aligned} G_i =& lb_i + r \times (ub_i - lb_i)\,, \\ i =& 1, 2, \dots N \end{aligned} \tag{5}$$

where, the individual $i$ within the search space has been depicted by $G_i$, and the lower an upper limitations of the individuals have been, in turn, displayed by $lb_i$ and $ub_i$.

- Phase 2: Candidate evaluation

The candidates' cost value within the population has been evaluated by employing a fitness function throughout the present phase. The fitness function has been utilized to measure a solution's quality because it assesses to what extent problem limitations are efficiently fulfilled.

- Phase 3: The present individuals have to be enhanced.

Once the review of the individuals' cost value has been accomplished, the update procedure of candidates gets started. The situations of candidates have been upgraded that is dependent on a step size variable, generating novel solution by the use of SCA. Then, novel solutions have been evaluated by employing the objective function, and the process has been recurred by the time the criterion of termination has been fulfilled.

- Phase 4: Assessing stage magnitude

By employing operator of stage magnitude ($\mathrm{St}_i$), the candidates have changed to the finest result within all iterations of the optimizer. There are three stages that have to be conducted for $St$ computation. The initial part illustrates the prior step size enhancement that has been multiplied by the operator of transform ($T$); moreover, it slightly gets decreased to make an optimal equilibrium between global search and local search ability of the optimizer.

Within the second stage, the distance between present situation and former best situation of a candidates has been ascertained by employing a parameter's cosine that is in the range [0, 1]. The final part illustrates the distance between candidate $i$ and the optimum situations that have been gained by all candidates.

Then, the aforementioned distance has been multiplied via a stochastic number's sine that is between 0 and 1. Throughout the initial stage of optimization process, $St_i$ is generated stochastically and enhanced after several iterations on the basis of the formula that has been illustrated in the following:

$$St_i\,(t+1) = T \times St_i\,(t) + A_1 \cos{(z_1)} \times (Gbest_i - X_i\,(t)) + A_2 \sin{(z_2)} \times (Gbest_i - X_i\,(t)) \tag{6}$$

here, the most optimal situation of the candidate $i$ within the present iteration has been illustrated by $Gbest_i$. Within the suggested approach, there are two stochastic number, $A_2$ and $A_1$, that are generated between 0 and 2. Moreover, two stochastically values, $z_2$ and $z_1$, are generated between 0 and 1. To enhance search efficacy throughout local search and global search stages, an operator of transfer $T$ has been employed, adjusting the search technique for regulation of local search and global search. The variable $T$ has been found to be a function that gets deteriorated after a while, which has been calculated in the following way[18]:

$$T = 100 \times \exp\left(\frac{-20t}{T_M}\right) \tag{7}$$

here, the highest quantity of iterations has been illustrated by $T_M$.0

- Phase 5: Constraint for magnitude of phase

The optimizer tried to adjust the distance of the candidate's motion within each dimension of the issue region throughout all iterations. The magnitude of the phase, which is a stochastic parameter that is able to make the candidates have movements in bigger cycles within the issue region, has been restricted via a particular range to prevent from explosion and divergence. It has been called constraint for magnitude of phase, which has been employed for controlling the motion of candidate and preventing them from going wider cycles within the issue space. Moreover, it ensures that the optimizer makes convergences toward the optimum solution. the present procedure can be thoroughly illustrated subsequently:

$$St_i \in [-St_{imin}, St_{imax}] \tag{8}$$

here, the lowest and the highest restriction of the motion value have been, in turn, depicted by $-St_{imin}$ and $St_{imax}$. These boundaries affect the movement of a candidate in its positioning arrangements throughout each iteration. The boundary can be expressed subsequently:

$$St_{imax} = 0.1 \times (ub_i - lb_i) \tag{9}$$

- Phase 6: renewal of location

Within the present phase, a novel candidate has been made for exploring the global optimal situation by the use of Eq. (10) that enables the optimizer to move to the optimum problem solver within solution space:

$$G_i\left(t+1\right) = G_i\left(t\right) + St_i(t+1) \tag{10}$$

## Fractional ordered-based golden search algorithm

The Golden Search Algorithm is known for its successful optimization outcomes, which has encountered challenges in certain scenarios. In this research, a new modification has been proposed to address these issues, offering more accurate and widespread results. This modification involves the use of fractional calculus (FC).

The fractional-order calculus (FC) is a valuable tool for enhancing the performance of meta-heuristic algorithms. The FC approach provides a well-organized consideration of process, memory, and inherent characteristics, making it a useful tool for improving the efficiency of meta-heuristic algorithms by taking into account memory aspects during solution updates. One of the popular models of FC is the Grunwald–Letnikov (GL) method which is defined in the following:

$$S^{\sigma}\left(G_i\left(t\right)\right) = \lim_{h\to 0}\frac{1}{h^{\sigma}}\sum_{a=0}^{\infty}(-1)^{a}\left(\begin{array}{c}\sigma\\ a\end{array}\right)G_i\left(t-ah\right) \tag{11}$$

where

$$\binom{\sigma}{a} = \frac{\Gamma(\sigma+1)}{\Gamma(a+1)\Gamma(\sigma-a+1)} = \frac{\sigma\left(\sigma-1\right)\left(\sigma-2\right)\ldots\left(\sigma-a+1\right)}{a!} \tag{12}$$

where $\Gamma\left(t\right)$ specifies the gamma function, and $D^{\sigma}\left(G_i\left(t\right)\right)$ represents the GL fractional derivative of order $\sigma$ which can be formulated as follows:

$$S^{\sigma}\left[G_i\left(t\right)\right] = \frac{1}{T^{\sigma}}\sum_{a=0}^{N}\frac{(-1)^{a}\,\Gamma\left(\sigma+1\right)G_i\left(t-aT\right)}{\Gamma\left(a+1\right)\Gamma\left(\sigma-a+1\right)} \tag{13}$$

where $\sigma$ signifies the derivative order operator, $T$ represents the sampling time, and $N$ specifies the length for memory. By assuming $\sigma = 1$, the previous equation has been reformulated as follows:

$$S^{1}\left[G_i\left(t\right)\right] = G_i\left(t+1\right) - G_i\left(t\right) \tag{14}$$

where $S^{1}\left[G_i\left(t\right)\right]$ defines the variance in the following. This study uses FC memory to update the renewal of location in phase 6, i.e.,

$$G_i\left(t+1\right) - G_i\left(t\right) = St_i(t+1) \tag{15}$$

So, the joint equation can be considered as follows:

$$S^{\sigma}\left[G_i\left(t+1\right)\right] = G_i\left(t\right) + \sum_{a=1}^{m}\frac{(-1)^{a}\,\Gamma\left(\delta+1\right)Z_i\left(t+1-a\right)}{\Gamma\left(a+1\right)\Gamma\left(\sigma-a+1\right)} = St_i(t+1) \tag{16}$$

By using the above assumptions, the renewal of location has been reformulated as follows:

$$G_i\left(t+1\right) = -\sum_{a=1}^{m}\frac{(-1)^{a}\,\Gamma\left(\sigma+1\right)G_i\left(t+1-a\right)}{\Gamma\left(a+1\right)\Gamma\left(\sigma-a+1\right)} + St_i(t+1) \tag{17}$$

Setting $m$ equal to 4 and considering first four terms of memory data, the renewal of location has been reformulated as follows:

$$\begin{aligned} G_i\left(t+1\right) = {} & \frac{1}{1!}\sigma G_i\left(t\right) + \frac{1}{2!}\sigma\left(1-\sigma\right)G_i\left(t-1\right) + \frac{1}{3!}\sigma\left(1-\sigma\right)\left(2-\sigma\right)G_i\left(t-2\right) \\ & + \frac{1}{4!}\sigma\left(1-\sigma\right)\left(2-\sigma\right)\left(3-\sigma\right)G_i\left(t-3\right) + St_i\left(t+1\right) \end{aligned} \tag{18}$$

## Results and discussions

In this study, the method analysis has been evaluated during some subsections which are explained in the following.

## Authentication of the proposed algorithm

To assess the effectiveness of the Fractional ordered-based Golden Search (FO-GS) algorithm, we conducted a comprehensive evaluation using the well-known CEC-BC-2017 test case. In this study, we compared the performance of the FO-GS algorithm against six prominent optimization algorithms: Multi-verse optimizer (MVO)[19], Owl Search Algorithm (OSA)[20], Squirrel search algorithm (SSA)[21], Billiard-based Optimization Algorithm (BOA)[22], World Cup Optimization (WCO)[23], and Biogeography-Based Optimizer (BBO)[24]. By benchmarking these algorithms, we aim to highlight the strengths and advantages of the FO-GS algorithm in solving complex optimization problems. The parameter configurations utilized by the algorithms employed are shown in Table 1.

To enable a meaningful comparison, a standardized parameter settings should be employed for all algorithms. The highest number of epochs and population size for all methods are consistently set at 200 and 60, respectively. To guarantee precise and dependable results, each technique was executed individually 15 times across all benchmark functions. The research used functions with a solution range from − 100 to 100, each characterized by ten dimensions. The evaluation results, comparing the FO-GS algorithm with multiple metaheuristic algorithms on the CEC-BC-2017 test functions, are detailed in Table 2.

Based on the evaluation results presented in Table 3, it is evident that the Fractional ordered-based Golden Search (FO-GS) algorithm outperforms the other six prominent optimization algorithms in terms of effectiveness and efficiency. When comparing the average (Avg) values across different functions, the FO-GS algorithm consistently achieves lower values compared to its competitors.

This indicates that FO-GS finds better solutions with lower function values, suggesting its superior performance in optimizing the given test functions. Furthermore, the FO-GS algorithm demonstrates higher stability and reliability compared to the other algorithms, as evident from the low standard deviation (StD) values. The StD values represent the variation or spread of the results obtained across multiple runs. Lower StD values indicate that the algorithm consistently finds good solutions, with less variation in the obtained results.

## WildWood optimization results

The results obtained after utilizing the Fractional Order-based Golden Search Algorithm to find the optimal hyperparameters are as follows:

The data suggests that the WildWood algorithm achieves optimal performance with a moderate tree count of 150 and a maximum tree depth of 8. The minimum samples per leaf is established at 5, indicating the algorithm's capability to manage smaller datasets effectively. A learning rate of 0.1 is employed, representing a balanced approach between exploration and exploitation. Lastly, a regularization strength of 0.01 is applied, implying that the algorithm incorporates a moderate degree of regularization to mitigate the risk of overfitting. The WildWood model has been retrained using optimal hyperparameters, leading to the computation of the cost function value. Table 4 indicates the optimal values achieved for the cost function of the suggested WildWood.

The optimization process has identified the best combination of hyperparameters for the WildWood algorithm for our purpose. With these optimal values, the model achieves an MSE of 0.85, indicating a good fit

| Algorithm | Set parameter | Value |
|---|---|---|
| MVO[19] | $WEP_{min}$ | 0.2 |
| | $WEP_{max}$ | 1 |
| | $Coefficient(P)$ | 6 |
| OSA[20] | $T_{dead}$ | 18 |
| | $\|P\|$ | 10 |
| | $Acc_{low}$ | 0.2 |
| | $Acc_{high}$ | 1 |
| SSA[21] | $N_{fs}$ | 4 |
| | $G_c$ | 1.9 |
| | $P_{dp}$ | 0.1 |
| BOA[22] | No. of pockets | 22 |
| | $w$ | 0.7 |
| | $ES$ | 0.3 |
| BBO[24] | Habitat modification probability | 1 |
| | Immigration probability | 1 |
| | Step size | 1 |
| | Max immigration (I) | 1 |
| | Max emigration (E) | 1 |
| | Mutation probability | 0.005 |
| WCO[23] | Play off | 0.04 |
| | ac | 0.3 |

**Table 1.** The parameter configurations utilized by the algorithms employed.

| Function | Indicator | FO-GS | MVO | OSA | SSA | WCO | BOA | BBO |
|---|---|---|---|---|---|---|---|---|
| F1 | Avg | 1.136443 | 25.117 | 13.757 | 21.493 | 54.793 | 21.863 | 2.571 |
| | StD | 0 | 2.353 | 7.729 | 13.038 | 6.726 | 0.981 | 2.952 |
| F3 | Avg | 0.785692 | 1.194 | 81.122 | 9.130 | 1.890 | 37.308 | 10.458 |
| | StD | 0 | 4.515 | 0.000 | 3.421 | 0.701 | 5.336 | 0.000 |
| F5 | Avg | 15.18834 | 32.369 | 206.251 | 262.964 | 75.514 | 38.111 | 218.862 |
| | StD | 0.114575 | 1.411 | 1.939 | 1.087 | 1.049 | 1.238 | 2.366 |
| F7 | Avg | 8.236361 | 13.739 | 21.371 | 115.681 | 29.148 | 174.250 | 124.714 |
| | StD | 0.028365 | 1.060 | 4.333 | 0.664 | 1.002 | 1.701 | 0.854 |
| F9 | Avg | 5.030082 | 159.255 | 54.172 | 199.840 | 182.184 | 120.450 | 149.494 |
| | StD | 0.006627 | 0.008 | 0.038 | 0.187 | 6.335 | 0.126 | 0.010 |
| F11 | Avg | 11.53439 | 74.358 | 247.821 | 259.173 | 14.802 | 123.758 | 270.148 |
| | StD | 0.314032 | 1.949 | 1.424 | 2.326 | 4.122 | 3.353 | 0.589 |
| F13 | Avg | 816.7334 | 2239.488 | 3402.930 | 875.580 | 4406.976 | 3540.631 | 1092.331 |
| | StD | 27.15951 | 684.309 | 1904.409 | 451.525 | 2985.347 | 1111.973 | 950.676 |
| F15 | Avg | 8.655348 | 83.274 | 224.295 | 958.658 | 1010.131 | 17.321 | 356.682 |
| | StD | 6.918234 | 18.730 | 8.292 | 17.178 | 171.135 | 12.316 | 17.142 |
| F17 | Avg | 4.644907 | 5.803 | 63.007 | 236.727 | 352.005 | 29.187 | 55.628 |
| | StD | 4.470481 | 12.605 | 13.269 | 8.764 | 4.947 | 9.997 | 7.340 |
| F19 | Avg | 5.578051 | 6.615 | 958.396 | 300.327 | 1544.253 | 345.579 | 1143.791 |
| | StD | 5.253497 | 11.784 | 358.223 | 329.076 | 996.747 | 12.640 | 402.171 |

**Table 2**. The evaluation results, comparing the FO-GS algorithm with multiple metaheuristic algorithms on the CEC-BC-2017 test functions.

| Hyperparameter | Optimal Value |
|---|---|
| Number of trees | 150 |
| Maximum tree depth | 8 |
| Minimum samples per Leaf | 5 |
| Learning rate | 0.1 |
| Regularization strength | 0.01 |

**Table 3**. The optimal values achieved for the parameters of the suggested *WildWood*.

| Cost function | Optimal value |
|---|---|
| Mean squared error (MSE) | 0.85 |

**Table 4**. The optimal values achieved for the cost function of the suggested *WildWood*.

between the predicted and actual traffic flow data. By tuning the hyperparameters, we have improved the model's performance and reduced the error in traffic flow predictions. The selected values strike a balance between model complexity and generalization, leading to more accurate and reliable predictions for smart city traffic management. This example demonstrates how cost function selection and hyperparameter optimization can enhance the effectiveness of the WildWood algorithm in addressing specific smart city challenges.

## Model analysis

In this section, the performance assessment of the suggested WildWood/FOGSA (Fractional Order-based Golden Search Algorithm) model was introduced and analyzed for the execution of smart cities. The assessment is carried out by contrasting WildWood/FOGSA with current optimization algorithms, including genetic algorithm (SGA)[25], improved ant colony optimization-simulated annealing (ACOSA)[25], and the original Golden Search Algorithm (GSA), across a range of performance indicators. These indicators consist of latency, energy usage, throughput, and network longevity.

*Latency*

Latency is the duration between a user's request and the server's response. It plays a crucial role in evaluating the system's responsiveness and effectiveness. Latency, denoted as L, is defined as:

$$L = T_{response} - T_{request} \tag{19}$$

where $T_{request}$ and $T_{response}$ are the time at which the user sends a request, and the time at which the server responds, respectively.

Figure 2 illustrates the latency comparison between conventional methods (SGA, ACOSA, and GSA) and the innovative WildWood/FOGSA approach.

The analysis demonstrates that the WildWood/FOGSA method exhibits the lowest latency compared to all other techniques, signifying its superior efficiency regarding computational time. In contrast, traditional methods such as SGA, ACOSA, and GSA show elevated latency levels, with SGA recording the highest latency. The reduction in latency associated with the WildWood/FOGSA method can be ascribed to the proficient optimization of hyperparameters facilitated by the Fractional Order-based Golden Search Algorithm (FOGSA). This algorithm effectively identifies the optimal hyperparameters, leading to a more efficient and rapid convergence of the WildWood algorithm. This suggests that WildWood/FOGSA is prompter and more effective in managing user requests.

*Energy consumption*

Efficient energy management is another important task in scheduling and IoT device management. It quantifies the energy utilized by individual nodes for data transmission or task completion. Energy Consumption, denoted as E, is defined as:

$$E = \sum_{i=1}^{N} P_i \times T_i \tag{20}$$

where $P_i$ describes the power consumption of the $ith$ node, and $T_i$ determines the time duration of its activity. Figure 3 provide a bar plot representation of the energy consumption comparison.

In comparison, traditional methods demonstrate greater energy consumption levels, with GSA utilizing 90 Joules, ACOSA at 92 Joules, and SGA exhibiting the highest energy usage at 98 Joules. This suggests that the WildWood/FOGSA method is the most energy-efficient of the four approaches. The reduction in energy consumption associated with the WildWood/FOGSA method can be credited to the effective optimization of hyperparameters through the Fractional Order-based Golden Search Algorithm (FOGSA). By refining the hyperparameters, the WildWood algorithm achieves enhanced performance while requiring fewer computations, thereby leading to decreased energy consumption. This demonstrates that WildWood/FOGSA is more energy-efficient, a crucial factor in extending the longevity of IoT devices and cutting down on operational expenses.

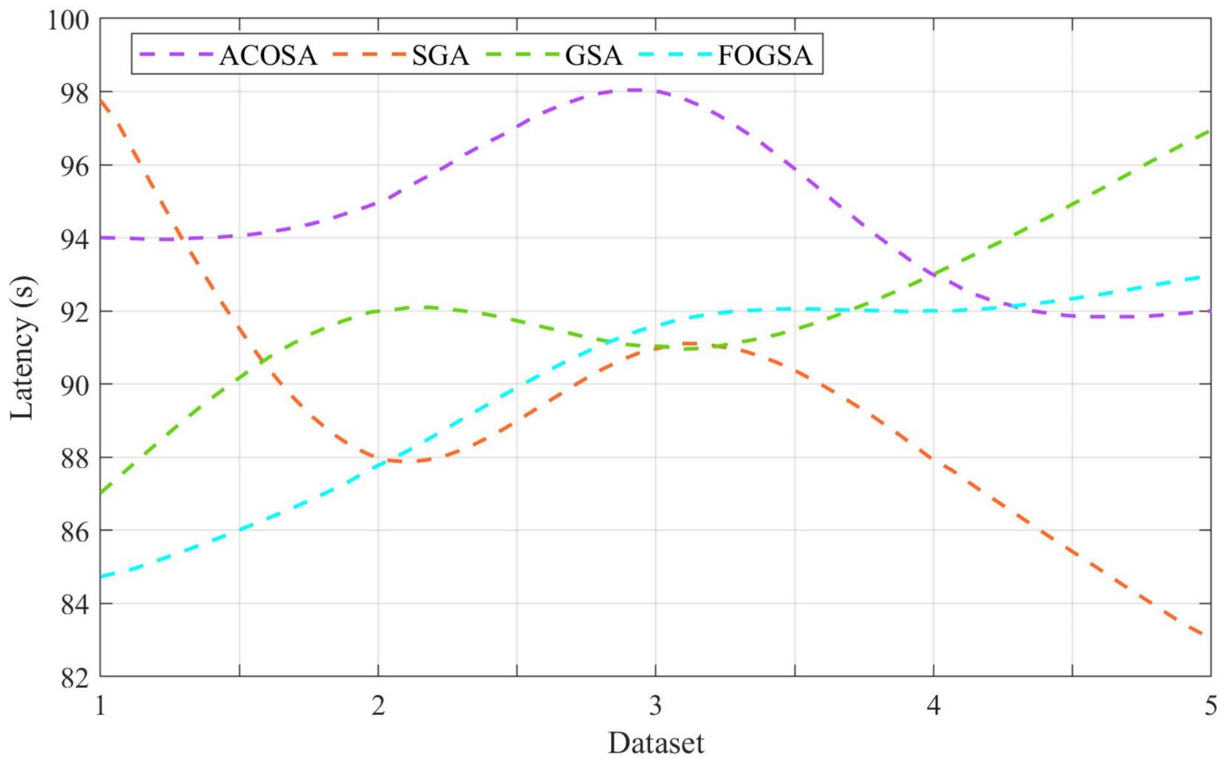


**Fig. 2**. Latency comparison between the conventional and proposed approaches.

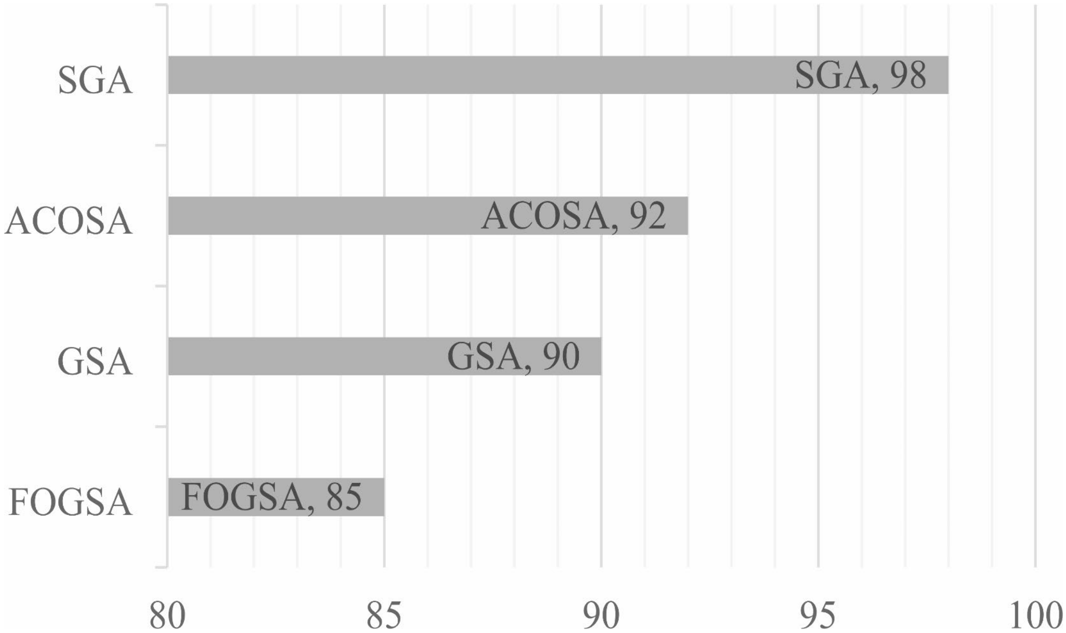


**Fig. 3**. A bar plot representation of the energy consumption comparison.

*Throughput*
Throughput is a significant metric that quantifies the volume of data that a system can handle or transfer during a specific period. It serves as a fundamental gauge of the system's performance and capability.

Throughput, denoted as T, is defined as:

$$T = \frac{Total\ data\ processed/transmitted\,(bits)}{Time\ duration\,(seconds)} \tag{21}$$

Figure 4 displays the comparison of throughput between the proposed WildWood/FOGSA model and the other state of the art models.

As shown in Fig. 4, WildWood/FOGSA model achieves a significantly higher throughput of 98% compared to the traditional methods. This demonstrates the superior efficiency and data processing capabilities of the proposed method.

*Network lifetime*
The network lifetime is the period until the initial node in the network depletes its energy, which is a crucial measure for evaluating the system's durability and viability. Lifetime, denoted as LT, is defined as:

$$LT = \min_{i=1,2,\ldots,N}\left(\frac{E_i}{P_i}\right) \tag{22}$$

where, $E_i$ and $P_i$ represent the initial energy of the $ith$ node, and its power consumption rate, respectively.

Figure 5 displays the comparison of lifetime between the proposed WildWood/FOGSA model and the other state of the art models.

The findings from Fig. 5 reveal that WildWood/FOGSA greatly enhances the network's durability, surpassing other methods. This underscores the efficiency of the proposed technique in managing energy consumption and extending the network's operational lifespan.

## Discussions
The results presented in this section highlight the exceptional performance of the WildWood/FOGSA optimization method for smart city implementation. WildWood/FOGSA consistently outperforms traditional methods (SGA, ACOSA, and GSA) across all metrics, including energy consumption, latency, throughput, and network lifetime. The improved performance of WildWood/FOGSA can be attributed to its innovative combination of WildWood optimization and the Fractional Order-based Golden Search Algorithm. This hybrid approach enabled efficient exploration of the solution space, leading to more effective decision-making and resource allocation in smart cities. By achieving lower latency, reduced energy consumption, higher throughput, and extended network lifetime, WildWood/FOGSA showed its potential to enhance the overall efficiency, sustainability, and reliability of IoT-enabled smart city systems. Furthermore, the results underscore the importance of optimizing hyperparameters and cost functions, as discussed earlier. By tailoring the WildWood

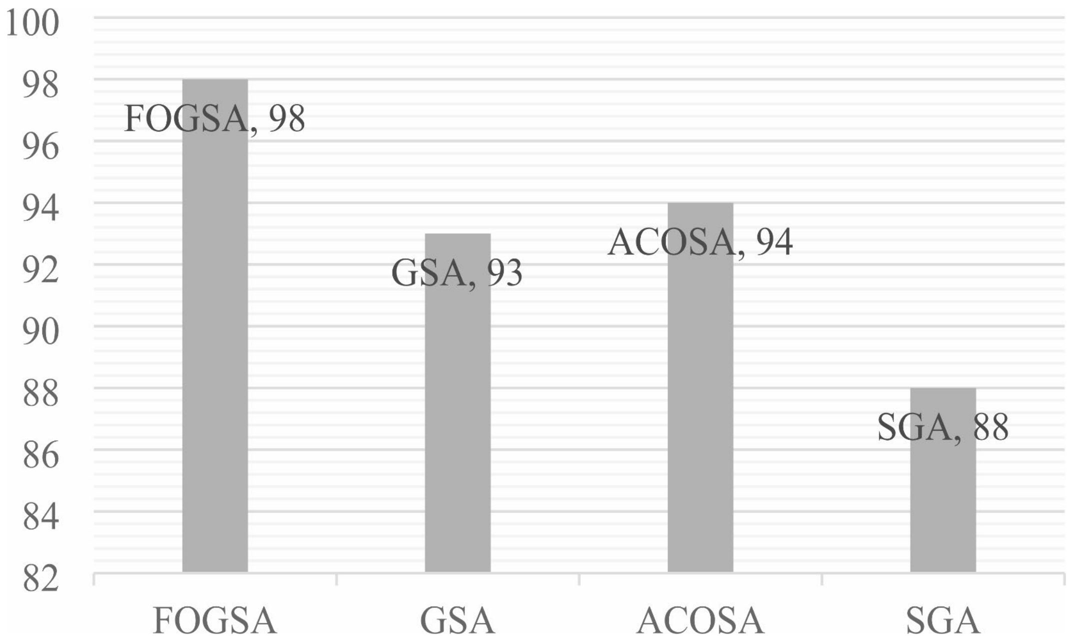


**Fig. 4**. Comparison of throughput between the proposed WildWood/FOGSA model and the other state of the art models.

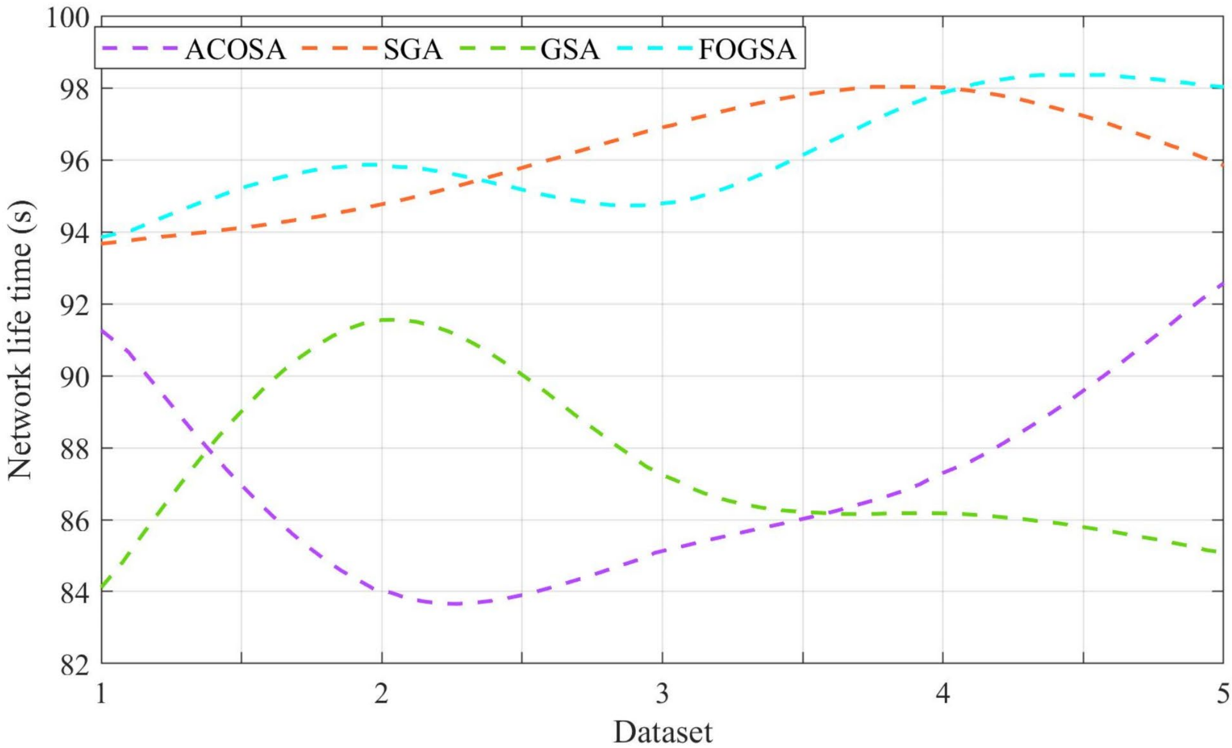


**Fig. 5**. Network lifetime comparison between the conventional and proposed approaches.

algorithm to the specific requirements of smart city applications, its performance and adaptability can be further enhanced to diverse scenarios.

The model participates the WildWood algorithm with the Fractional Order Golden Search Algorithm to0 make it applicable in different real-world Internet of Things (IoT) settings, such as smart cities, industrial automation, agriculture, and healthcare. This integration aims to enhance the performance of IoT devices across systems like traffic management, energy management, predictive maintenance, patient monitoring, and crop monitoring.

Nonetheless, the implementation of our model in actual IoT networks faces several limitations and challenges, including issues related to scalability, data quality, security, interoperability, real-time processing, energy efficiency, and network latency. To address these challenges, we recommend employing distributed computing, robust security measures, data preprocessing techniques, interoperability standards, real-time processing methodologies, energy-efficient algorithms and hardware, as well as network optimization strategies. By attempting these obstacles, the model can be successfully deployed in practical IoT environments, ensuring optimized performance and effective management of IoT devices.

Also, alongside all the advantages of the proposed method, there are some limitations that can be considered in the future work. Firstly, it requires substantial amounts of training data to effectively optimize hyperparameters, which poses a challenge in smart city applications where data availability may be restricted. Moreover, the framework is prone to overfitting, particularly when handling complex and diverse datasets. The use of the Fractional Order Golden Search Algorithm within the framework can also lead to significant computational complexity, thereby increasing the time and resources needed for processing. Additionally, the framework's performance is highly sensitive to the selection of hyperparameters. Lastly, the applicability of the proposed framework may be constrained, as it may not generalize effectively to other smart city applications or scenarios.

## Conclusions

The growing of smart cities cruces on the ability to extract meaningful insights from the ever-increasing volume of data generated by ubiquitous Internet of Things (IoT) sensors. This data holds immense potential to optimize city operations, improve resource allocation, and ultimately enhance the quality of life for residents. However, effectively utilizing this data requires robust techniques for both data analysis and model optimization. This study explored the application of machine learning, specifically the WildWood algorithm, for analyzing data collected from various IoT sensors within a smart city environment. WildWood's strength lies in its ability to handle complex and potentially noisy data, making it well-suited for this task. However, optimizing the performance of machine learning models is crucial for ensuring accurate and reliable results. To address this, a novel approach was proposed using the Fractional Order Golden Search Algorithm for hyperparameter optimization. This optimization technique offers advantages like avoiding local optima and potentially achieving faster convergence compared to traditional methods. The simulations demonstrated the effectiveness of the proposed framework. The WildWood model achieved significantly improved efficiency when optimized with the Fractional Order Golden Search Algorithm. This translates to more accurate and reliable insights gleaned from smart city data, empowering city officials to make data-driven decisions that benefit citizens. Future work could focus on evaluating the proposed framework on real-world smart city datasets with various sensor types, comparing the performance of WildWood with other machine learning algorithms suitable for smart city applications, investigating the integration of alternative optimization algorithms with WildWood and exploring their potential benefits, and unlocking the full potential of smart city data, by continuing to explore and refine analytical approaches leading to a more efficient, sustainable, and citizen-centric urban future.

## Data availability

All data generated or analysed during this study are included in this published article.

Received: 23 July 2024; Accepted: 13 November 2024
Published online: 26 November 2024

## References

1. Rajyalakshmi, V. & Lakshmanna, K. Intelligent face recognition based multi-location linked IoT based car parking system. *IEEE Access* **11**, 84258–84269 (2023).
2. Liu, Y. & Bao, Y. Intelligent monitoring of spatially-distributed cracks using distributed fiber optic sensors assisted by deep learning. *Measurement* **220**, 113418 (2023).
3. Rajyalakshmi, V. & Lakshmanna, K. Detection of car parking space by using hybrid deep densenet optimization algorithm. *Int. J. Netw. Manag.* **34**(1), e2228 (2024).
4. Zhou, J. et al. A lightweight energy consumption ensemble-based botnet detection model for IoT/6G networks. *Sustain. Energy Technol. Assess.* **60**, 103454 (2023).
5. Alqahtani, F., Al-Makhadmeh, Z., Tolba, A. & Said, W. Internet of things-based urban waste management system for smart cities using a Cuckoo Search Algorithm. *Cluster Comput.* **23**, 1769–1780 (2020).
6. Annadurai, C. et al. Biometric authentication-based intrusion detection using artificial intelligence internet of things in smart city. *Energies* **15**(19), 7430 (2022).
7. Chen, N., Qiu, T., Zhao, L., Zhou, X. & Ning, H. Edge intelligent networking optimization for internet of things in smart city. *IEEE Wirel. Commun.* **28**(2), 26–31 (2021).
8. Yang, G., Du, S., Duan, Q. & Su, J. Short-term price forecasting method in electricity spot markets based on attention-LSTM-mTCN. *J. Electric. Eng. Technol.* **17**, 1–10 (2022).
9. Cepeda-Pacheco, J. C. & Domingo, M. C. Deep learning and Internet of Things for tourist attraction recommendations in smart cities. *Neural Comput. Appl.* **34**(10), 7691–7709 (2022).
10. Vozvakhov, I. A., Strakhov, M. D., Ershov, V. A., Agapov, M. Y., Tatarinov, V. V. & Rozhok, A. "Moving objects tracking method based on discharged optical flow. In *2022 4th International Youth Conference on Radio Electronics, Electrical and Power Engineering (REEPE)*, 1–5 (IEEE, 2022).
11. Kolhe, R. V., William, P., Yawalkar, P. M., Paithankar, D. N. & Pabale, A. R. Smart city implementation based on Internet of Things integrated with optimization technology. *Meas. Sens.* **27**, 100789 (2023).

12. Razmjooy, N., Bahmanyar, D., Rajinikanth, V. & de Oliveira, G. G. A new home energy management system for smart home using improved multi-objective antlion optimization algorithm. In *Proceedings of the 7th Brazilian Technology Symposium (BTSym'21) Emerging Trends in Human Smart and Sustainable Future of Cities*, Vol. 1, 56–72 (Springer, 2022)
13. Sinsomboonthong, S. Performance comparison of new adjusted min-max with decimal scaling and statistical column normalization methods for artificial neural network classification. *Int. J. Math. Math. Sci.* **2022**(1), 3584406 (2022).
14. Singh, D. & Singh, B. Investigating the impact of data normalization on classification performance. *Appl. Soft Comput.* **97**, 105524 (2020).
15. Zhu, C. et al. Analytical study of nonlinear models using a modified Schrödinger's equation and logarithmic transformation. *Results Phys.* **55**, 107183 (2023).
16. Revetria, R., Damiani, L., Ivanov, M. & Ivanova, O. An hybrid simulator for managing hydraulic structures operational modes to ensure the safety of territories with complex river basin from flooding. In *2017 Winter Simulation Conference (WSC)*, 2717–2728 (IEEE, 2017)
17. Gaïffas, S., Merad, I. & Yu, Y. WildWood: A new random forest algorithm. *IEEE Trans. Inform. Theory* **69**(10), 6586–6604 (2023).
18. Noroozi, M. et al. Golden search optimization algorithm. *IEEE Access* **10**, 37515–37532 (2022).
19. Mirjalili, S., Mirjalili, S. M. & Hatamlou, A. Multi-verse optimizer: A nature-inspired algorithm for global optimization. *Neural Comput. Appl.* **27**(2), 495–513 (2016).
20. Jain, M., Maurya, S., Rani, A. & Singh, V. Owl search algorithm: A novel nature-inspired heuristic paradigm for global optimization. *J. Intell. Fuzzy Syst.* **34**(3), 1573–1582 (2018).
21. Jain, M., Singh, V. & Rani, A. A novel nature-inspired algorithm for optimization: Squirrel search algorithm. *Swarm Evolut. Comput.* **44**, 148–175 (2019).
22. Kaveh, A., Khanzadi, M. & Moghaddam, M. R. Billiards-inspired optimization algorithm; a new meta-heuristic method. In *Structures* vol 27, 1722–1739 (Elsevier, 2020).
23. Razmjooy, N., Khalilpour, M. & Ramezani, M. A new meta-heuristic optimization algorithm inspired by FIFA world cup competitions: Theory and its application in PID designing for AVR system. *J. Control Autom. Electric. Syst.* **27**(4), 419–440 (2016).
24. Simon, D. Biogeography-based optimization. *IEEE Trans. Evolut. Comput.* **12**(6), 702–713 (2008).
25. Zhang, C. Design and application of fog computing and Internet of Things service platform for smart city. *Future Gener. Comput. Syst.* **112**, 630–640 (2020).


## Author contributions
All authors wrote the main manuscript text. All authors reviewed the manuscript.

## Funding
Ideological and political Teaching Project of higher Education courses in Zhejiang Province: Research on the ideological and political Teaching Mode of "non-Heritage Culture +" Art and Design course(Project number: SKCSZ-2022-A490).

## Declarations

## Competing interests
The authors declare no competing interests.

## Additional information
**Correspondence** and requests for materials should be addressed to Z.Z.

**Reprints and permissions information** is available at www.nature.com/reprints.

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.