

RESEARCH

Open Access



A new QoS-aware service discovery technique in the Internet of Things using whale optimization and genetic algorithms

Xiao Liu¹ and Yun Deng^{2*}

*Correspondence:
13257718870@163.com

¹ Department of Information Engineering, Guangxi Technological College of Machinery and Electricity, Nanning 530007, China

² School of Big Data, Guangxi Vocational & Technical College, Nanning 530226, China

Abstract

Rapid technological advances have made daily life easier and more convenient in recent years. As an emerging technology, the Internet of Things (IoT) facilitates interactions between physical devices. With the advent of sensors and features on everyday items, they have become intelligent entities able to perform multiple functions as services. IoT enables routine activities to become more intelligent, deeper communication, and processes more efficient. In the dynamic landscape of the IoT, effective service discovery is key to optimizing user experiences. A Quality of Service (QoS)-aware service discovery technique is proposed in this paper to address this challenge. Through whale optimization and genetic algorithms, our method aims to streamline decision-making processes in IoT service selection. The bio-inspired optimization techniques employed in our approach facilitate the discovery of services more efficiently than traditional methods. Our results demonstrate superior performance regarding reduced data access time, optimized energy utilization, and cost-effectiveness through comprehensive simulations.

Keywords: Service discovery, Internet of Things, Service selection, Optimization

Introduction

The Internet of Things (IoT) technology connects physical objects, computers, and humans [1, 2]. IoT has the potential to revolutionize how people interact with their environment, allowing for more efficient use of resources and improved automation. It can also create new business opportunities and provide consumers more convenience [3]. A unique IP address is assigned to every item under Internet Protocol (IP) for information exchange and communication, enabling intelligent positioning, tracking, monitoring, and management of items with RFID, infrared sensors, global positioning systems, laser scanners, and other information-sensing equipment [4]. This technology allows businesses to monitor and optimize their supply chain processes, improve customer service, and reduce operational costs. It also gives consumers real-time information about products, such as availability and pricing [5]. The system architecture may, for example, be determined by the context in which operations and processes are carried out in

real-time. For example, an electrical switch box in a smart home could be controlled remotely by a smartphone [6].

IoT systems are characterized by their capacity for selecting and combining services under end users' requirements and preferences. This means that the systems can be customized to the user's needs, allowing them to select the most important services and combine them in a most efficient way [7]. Service selection forms the basis of service composition, which involves determining appropriate services among independent candidates to meet the end user's needs. These candidate services may be functionally equivalent; however, their non-functional characteristics may differ. The non-functional characteristics are quality of service (QoS) parameters with varying values [8]. It is, therefore, imperative to consider QoS factors that align with the end user's needs. Smart devices are becoming increasingly prevalent, generating and publishing many Internet-based services. Public transportation facilitates IoT service discovery by serving as a dynamic and interconnected network that facilitates the seamless integration and accessibility of diverse services, contributing to enhanced mobility and efficiency in smart city ecosystems [9].

Nevertheless, context-aware environments do not require all services. In most cases, accessing a particular set of services at a specific time is necessary to meet certain needs. A service discovery function allows a network to search effectively for devices providing the required services. A critical role for service discovery in IoT is highlighted in [7]. This functionality, however, poses several challenges because an excessive number of devices would result in large search areas. In addition, network traffic may become too large to handle efficiently, resulting in scalability concerns. The service discovery problem is also confronted with other issues, such as environmental context, geographical constraints, and dynamic availability. A distributed indexing and searching architecture is needed to overcome these challenges to support efficient service discovery. This architecture should be able to manage large-scale networks, scale to meet changing demand, and consider environmental context [10].

Integrating meta-heuristic algorithms, machine learning, deep learning, and neural networks is paramount in advancing the domain of IoT service selection and discovery [11]. As the IoT ecosystem expands, interconnected devices' sheer complexity and scale necessitate intelligent and adaptive methodologies for efficient service discovery. Meta-heuristic algorithms, such as the Whale Optimization Algorithm (WOA), Simulated Annealing (SA), and Imperialists Competitive Algorithm (ICA), offer robust optimization techniques that navigate the vast solution space to enhance the accuracy and speed of service selection [12]. Machine learning techniques, coupled with deep learning and neural networks, bring a transformative dimension by enabling systems to autonomously learn and adapt to evolving patterns within the IoT network [13, 14]. These technologies empower service discovery mechanisms to comprehend intricate relationships, dependencies, and dynamic changes in real-time, thereby optimizing decision-making processes [15]. Through the utilization of these advanced algorithms, IoT environments can achieve enhanced QoS, reduced latency, and improved resource utilization [16]. The synergy of meta-heuristic algorithms and machine learning paradigms streamlines the service discovery process and fosters adaptability and intelligence in IoT systems, ensuring they remain responsive to the dynamic and diverse demands of modern

interconnected environments [17, 18]. As we continue to explore the frontiers of IoT innovation, incorporating these technologies stands as a cornerstone, shaping the evolution of intelligent and efficient service selection in the IoT landscape [19, 20].

The selection of services becomes even more challenging when combined with specific requirements for quality of service. When dealing with this situation, it is important to select the most appropriate service that satisfies the end user's functional and quality requirements, not simply identify, select, and combine services with specific capabilities [21]. Optimizing algorithms are required to select services consistent with specific criteria, including energy consumption, response time, reliability, availability, cost, throughput, and accuracy. In recent years, research has focused on developing algorithms for optimizing the selection of IoT services based on a defined set of quality requirements. The most well-known algorithms are cluster head selection, shortest path algorithms, mapping flow-based algorithms, co-locating services, physical services models, genetic algorithms (GAs), and particle swarm optimization (PSO) algorithms.

Related work

NWE, et al. [22] proposed an adaptive QoS-based service selection technique that allows users to provide subjective preferences in a friendly and simple manner. It identifies services under subjective information given by service consumers and objective information provided by service providers. Using artificial neural network backpropagation, it finds objective factors and evaluates the creditability of user subjective factors using similarity aggregation, which has already been evaluated using ontological reasoning through the proposed Quality of Service ontology. According to the comparative study and experimental results, the proposed algorithm outperforms other methods of selecting services.

Jia, et al. [23] presented a service discovery method that employs a multi-stage semantic service matching strategy. It uses a layer filtering approach, considering various constraints associated with IoT services, including quality of experience, precondition/effect, input/output, and service category. It obtains accurate matching results more efficiently. In the first place, they describe IoT services and requests uniformly using OWL-Siot, which is an IoT service description language. Then, a 4-layer structure model for service discovery is proposed, comprising a first layer for interactive interfaces, a second layer for parsing annotations, a third layer for service matching, and a fourth layer for data semantics. A hybrid service matching degree measurement is also proposed by synthesizing the concept logic and semantic similarity on a layer-by-layer basis. In experimental studies, the method is effective in improving service discovery performance.

Osamy et al. [24] presented a novel adaptive service discovery architecture to enable service discovery and service selection in IoT-based sensor networks with a high degree of dynamicity. It offers a robust framework to provide user-oriented and environment-aware service discovery. The proposed architecture is based on a multi-agent system and can provide a dynamic, distributed, and self-organized service discovery architecture. It also offers a reliable and efficient search mechanism for selecting services most appropriate for the user's requirements and the environment. An extensive simulation is conducted to evaluate the proposed architecture. The experimental results demonstrate the

superiority of the proposed architecture over existing baselines regarding query success rate, number of transmission messages, and transmission cost.

Pattar et al. [25] proposed a progressive search algorithm for mapping user requirements to IoT resources and smart services, thereby providing personalized search results. Users' requirements have been divided into essential and optional requirements, which in turn have been correlated with the intrinsic and extrinsic properties of smart services. To refine the search results, two search strategies are suggested, namely, primitive search and elaborate search. In order to establish the practicability of the proposed search algorithm, empirical studies were conducted on a Smart Airport ecosystem and compared with the existing approaches in the literature.

Bensalah Azizou et al. [26] proposed a new distributed service discovery technique based on the Ant Colony Optimization (ACO) algorithm. Their technique uses the ACO algorithm to find the shortest path from the service consumer to the service provider. The algorithm uses the probabilistic behavior of ants in the natural environment to identify the most efficient paths to the destination. Simulation results indicate that the proposed approach performs well with fewer hops used to discover services and can achieve a high discovery success rate. The proposed approach also offers load balancing and can easily be adapted to dynamic environments.

Jia et al. [27] developed a privacy-aware service discovery technique using the artificial fish swarm algorithm. This method strives to select services that offer an optimal Quality of Experience (QoE), incorporating privacy requirements, to minimize privacy concerns and select services that meet all user requirements. As a first step, a model of quality of experience based on privacy requirements is created, and restrictions along with quantitative measures are presented. The second step is constructing an algorithm that utilizes the above model to determine which services should be selected. As a final step, the suggested strategy is validated by simulation. According to the results, this algorithm performed better in terms of accuracy and recall than the GA-based algorithm, demonstrating that it can effectively and feasibly solve privacy-sensitive service selection problems with IoT effectively and feasibly.

Zhu et al. [28] proposed a novel access selection approach that facilitates users to request IoT services in heterogeneous 5G networks while achieving optimal quality of experience. Specifically, a fuzzy logic-based network pre-assessment process filters available networks by the user device. In addition, a framework for evaluating preferences is developed that employs the objective entropy weight method and a subjective analytical hierarchy process to determine preference levels. Then, they developed an algorithm based on relative entropy for selecting multi-service networks quickly from the filtered available networks to enhance the quality of service for users under the constraints of limited network capacity. According to the evaluation results, the proposed mechanism is effective at maintaining stable service connections and significantly improving the quality of service for users.

Methods

This section describes a process for discovering and selecting cloud-edge services based on factors affecting QoS for an IoT platform. Additionally, a general description, a model for discovering services, and a methodology for selecting services based on a hybrid

WOA-GA algorithm are provided. The process consists of four steps: (1) identifying the factors that affect QoS, (2) discovering cloud-edge services in the IoT platform, (3) modeling the discovery process, and (4) selecting services based on a hybrid WOA-GA algorithm.

Problem statement

In order to optimize the QoS parameters in IoT environments, we need to choose the appropriate automated services. Research on service discovery has primarily focused on two approaches: syntactic and semantic. In each method, the search is conducted in a different manner. Syntactic service discovery is based on identifying the features of a service and matching them to a predefined set of requirements. Semantic service discovery uses semantic annotations to match services to user needs. Both methods are used to optimize QoS in IoT environments. Semantic approaches are limited by their reliance on syntactic theories. In addition, many of these approaches have difficulties or cannot be applied in the environment. Therefore, a hybrid syntactic and semantic service discovery process is necessary to achieve optimal QoS in IoT environments. This hybrid approach can leverage the strengths of both syntactic and semantic service discovery to provide a more efficient and effective way of discovering and connecting services. We propose a system for retrieving syntactic information about services that use data collected from IoT devices to identify the best services.

In our method, clients or IoT nodes send their desired service specifications. Service providers, in this instance an IoT node, provide their QoS measurements and information about the services using Web Service Definition Language (WSDL), which is used by the evaluation system to determine whether the candidate IoT node is appropriate for the Service Requester (SR) requirements. The evaluator system will then compare the SR and the measured QoS and, based on the user-defined parameters, will determine whether the candidate IoT node is eligible for deployment. If the candidate node is chosen, the evaluator will inform the Client and the IoT node of acceptance. Figure 1 illustrates how a service search is conducted within an IoT network.

Procedure for collecting service data

The information that leads to the discovery of the service is derived through a single source but in variable forms and with additional details. This source can be the customer's request, the market research, or the customer feedback on the existing services. As more details become available, the scope of the service increases, and more elements are added. The end result is a comprehensive service that meets the customer's needs. Service discovery will primarily rely on QoS measurements sent from IoT devices; however, we will also use the information provided by IoT device users to select more appropriate matches to fulfill SR requirements. In addition, information about the Service Provider (SP) can be used in the discovery of services. In order to verify the QoS factors, the service matching broker exchanges information with the node data administrator. The service matching broker then queries the service provider's database and retrieves the required information. The node data administrator then compares the retrieved data with the QoS requirements and selects the best-matching service. Finally, the service is provided to the user.

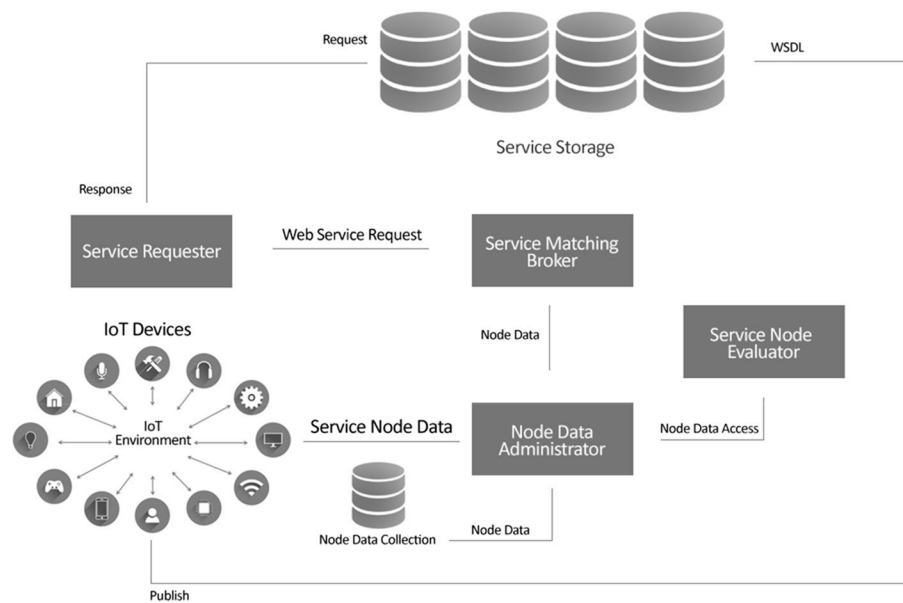


Fig. 1 A conceptual and operational overview of service discovery architecture

The IoT devices serve as service node data and send QoS measures to node data administrators. Node Data Administrators (NDAs) receive information from IoT devices directly and are prepared to act as service providers in the network. NDAs are able to monitor the performance of the network and make necessary changes for improved services. The data collected by the NDAs can also be used to improve the forecasting of future network performance. This helps to ensure a high quality of service for users of IoT devices. Data concerning QoS will be identified and forwarded inside the NDA to the Service Node Evaluator (SNE). In its QoS analyzer, SNE analyzes QoS measures. In the process of analyzing QoS, measurements generated by IoT devices will be examined and checked. This is to ensure that they satisfy service expectations within the network. The network administrator sets these requirements. Upon completion of the analysis of QoS measures, the information will be returned to NDA and placed in a database in order to preserve information regarding QoS.

The WSDL in our model collects information generated by IoT devices and user interactions periodically. The information we obtain in this way will be helpful in the initial discovery process by minimizing the number of candidates for service discovery and providing greater detail about the service provider. This information will be sent from IoT devices to service storage in the form of WSDL. This information will be stored in the service storage so the service matching broker can use it in response to an SR request. We will use this information to minimize potential candidates for SR further. Both NDA and QoS measures will be used to collect this data. In the same manner as QoS indicators, the information collected will be verified by an IoT information evaluator before it is available for inclusion in service storage and WSDL-based IoT information.

Procedure for discovering services

In order to find optimal automated services, we consider three QoS factors. As indicated in Eq. 1, three factors, including cost, execution time, and energy consumption, come together in order to determine the overall QoS value used for measuring the fitness function in optimal service discovery. The aim of this approach is to find the optimal services for a given user request with an acceptable trade-off between the three QoS factors. This enables users to obtain services with the best performance at an affordable cost. The objective of the fitness function is to minimize the overall QoS value, resulting in optimal automated services. The optimal services are then used to create a solution that meets the QoS requirements of the system. The Whale Optimization Algorithm (WOA) provides the best solution for discovering cloud-edge services.

The formal solution for achieving and determining efficient service discovery and IoT smart objects is formulated in the following manner. The IoT network consists of a variety of smart devices and sensors. SLA agreements specify that these smart applications rely on an array of services known as S at the edge layer and on a variable known as N at the sensor layer. Cloud-edge services typically outnumber sensors. The sensors also incorporate a limited power-saving feature, which is indicated by E_i . E^T shows the total energy consumed by a set of sensors. The proposed method defines the service capabilities of intelligent sensors and applications as follows:

$$E^T = \sum EN_1, EN_2, \dots, EN_M \quad (1)$$

In Eq. 1, S and A sets define the respective positions of guide sensors and items, respectively. Thus, a guide object can determine its position using the Global Locator; however, the significant problem lies in the lack of information regarding the location and position of the sensors, so it is necessary to determine their locations. The primary issue with this study is that it is assumed that three locator objects are defined as the positions of (a), but an object's position, such as $|x, y|$, remains unspecified. Thus, the following steps must be followed to identify this object's location.

- Calculate and estimate the distance between an unknown object and objects with known locations.
- The object's location is first estimated by using the location of three reference points.
- Several locations in the vicinity of the present position are derived from three sources, and those positions represent the WOA population. From there, multiple solutions are produced, and then a more precise location is attempted.

The WOA can be used to estimate distances between sensor objects based on three reference objects. The three-reference strategy allows us to obtain the object's position. Nonetheless, the three-reference method's locating error significantly depends on the accuracy with which the distance between the sensor object and the reference object is calculated. The WOA algorithm uses swarm intelligence to solve continuous optimization problems. WOA exhibits superior performance than recent meta-heuristic algorithms. Compared to other swarm intelligence algorithms, it is more straightforward and robust. The desired result can be achieved in practice by adjusting a single parameter (time interval). It involves humpback whales searching for food

in a multidimensional environment. Locations of humpback whales are considered decision variables, while distances between them and food are considered objective costs. Whales' time-dependent location is determined by three operational processes: searching for prey, using bubble nets to attack, and shrinking encircling prey. Figure 2 illustrates the primary presentation of the WOA.

The behavior of humpback whales in bubble nets can be described mathematically as follows.

$$\vec{X}(t+1) = \vec{D}' \cdot e^{bl} \cdot \cos(2\pi l) + \vec{X}^*(t) \tag{2}$$

$$\vec{X}(t+1) = \begin{cases} \vec{X}^*(t) - \vec{A} \cdot \vec{D} & \text{if } p < 0.5 \\ \vec{D}' \cdot e^{bl} \cdot \cos(2\pi l) + \vec{X}^*(t) & \text{if } p \geq 0.5 \end{cases} \tag{3}$$

In Eq. 3, p describes the logarithmic spiral's shape, and k refers to a uniformly distributed number. As a global optimizer, if $A > 1$ or $A < -1$, a randomly selected search agent replaces the best search agent as follows:

$$\vec{D} = |\vec{C} \cdot \vec{X}_{rand} - \vec{X}| \tag{4}$$

$$\vec{X}(t+1) = \vec{X}_{rand} - \vec{X} \cdot \vec{D} \tag{5}$$

where X_{rand} is randomly selected from whales in the current iteration.

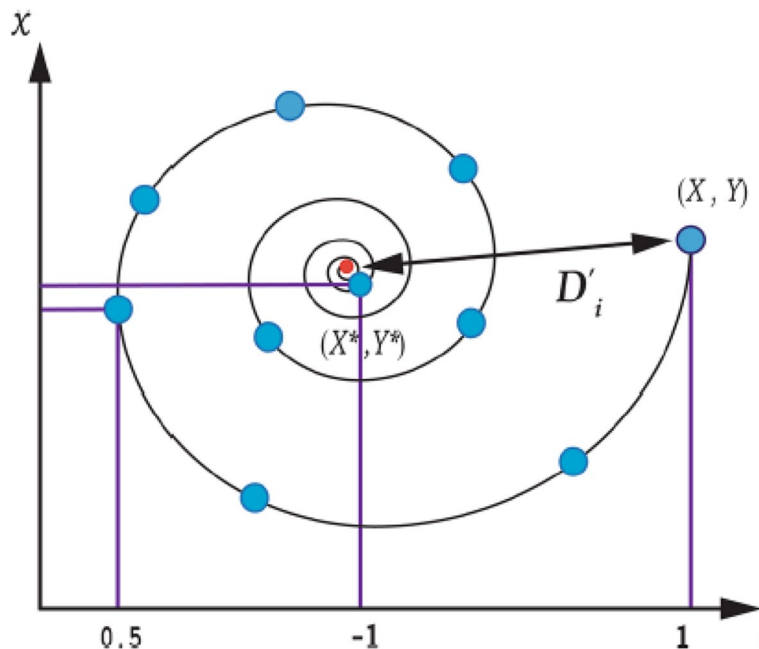


Fig. 2 Position update in a spiral

Service selection process

In order to select automated services, GA employs a Weighted Acyclic Graph (WAG) method to select discovered automated services. The discovered services are evaluated based on QoS factors. GA draws inspiration from Darwin's biological evolution theory. In fact, GA simulates natural selection. In nature, organisms with higher fitness exhibit a higher survival rate. This facilitates the transmission of their genes to future generations. Genes that allow species to better adapt to their environment (avoid enemies and find food) become dominant over time. GA models optimization problems by employing a set of variables derived from natural chromosomes and genes. In an optimization problem, each solution represents a chromosome, and each gene denotes a variable of the problem. As an example, GA utilizes chromosomes containing ten genes when dealing with a problem involving ten variables. GA enhances chromosomes in each generation through three main operations: selection, crossover, and mutation. In GA, a set of random chromosomes is used to begin the optimization process. Binary GA is based on the following equation:

$$X_i = \begin{cases} 1 & r_i < 0.5 \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

where r_i represents a random number between 0 and 1, independently calculated for each gene, and X_i denotes the i th gene. In continuous GA, genes are randomly initialized using the following equation:

$$X_i = (ub_i - lb_i) \times r_i + lb_i \quad (7)$$

where lb_i and ub_i correspond to the lower and upper bounds for the i th gene (variable), respectively. The primary goal is to obtain uniformly distributed random solutions over the entire set of variables for use in the following operations. Parents' characteristics should be reflected in future generations of the population while minimizing the possibility of selecting individuals with poor fitness values. It is first necessary to calculate the probability of selecting each individual, which can be calculated as follows:

$$P(i) = \frac{f(i)}{\sum_{i=1}^N f(i)} \quad (8)$$

where $f(i)$ represents the fitness value of an individual and $P(i)$ indicates the probability of the individual being selected. It is more likely that an individual with a higher fitness value will be selected than one with a lower fitness value. The crossover operator plays an important role in GA. It is required to perform the crossover operator between two individuals in order to improve global search capability. As a measure of crossover probability, we use the parameter P-crossover, expressed as follows:

$$P - crossover = k_1 \frac{f_{max} - f_a}{f_{max} - f_b} \text{ iff } f_a \geq f_b \quad (9)$$

$$P - crossover = k_2 \text{ iff } f_a \leq f_b \quad (10)$$

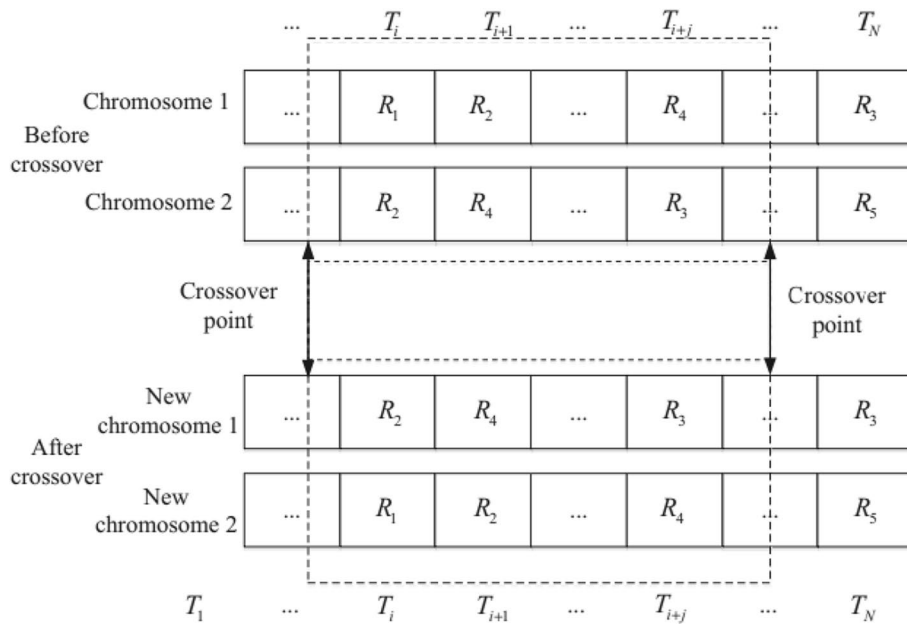


Fig. 3 Chromosome crossover operator

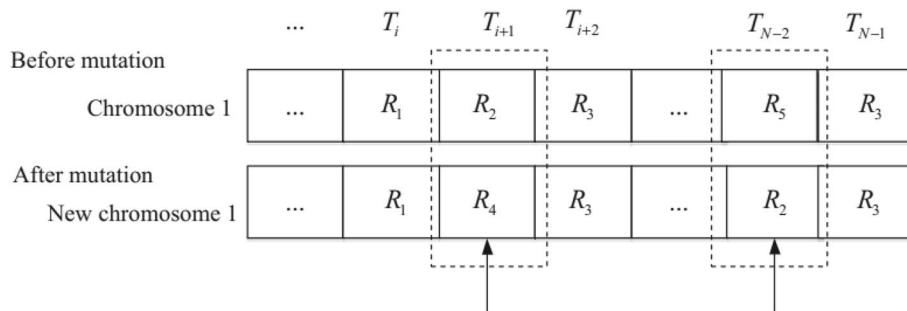


Fig. 4 Chromosome gene variation

where k_1 and k_2 share the same value between 0 and 1, f_b represents the average fitness value of the overall population, f_a represents the largest fitness value among the two individuals required to perform the crossover operator, and f_{max} reflects the maximum fitness value observed in a population. The crossover operator is shown in Fig. 3.

The mutation operator enhances GA's local search capabilities and maintains the population's diversity. Equations (5) and (6) provide formulae for calculating the mutation probability. The crossover operator is illustrated in Fig. 4.

$$P - mutate = k_3 \frac{f_{max} - f_c}{f_{max} - f_b} \text{ iff } f_c \geq f_b \tag{11}$$

$$P - mutate = k_c \text{ iff } f_c < f_b \tag{12}$$

As depicted in Fig. 4, two chromosome genes can be affected separately at locations $(i + 1)$ and $(N - 2)$. At location $(i + 1)$, the gene value is between R_2 and R_4 . Gene

values range from $R5$ to $R2$ at locations $(N-2)$. Mutations in genes may lead to positive outcomes. Sometimes, it may result in negative outcomes. Nevertheless, maintaining genetic diversity within a population requires the mutation of genes.

The proposed method begins by randomly selecting an initial server from the alternative servers, designating it as the first in the selected population. If the chosen server hosts the requested service, it is added to the candidate server set. Subsequently, another server is randomly chosen from the nodes linked to the initial server, becoming the second in the population and the new current server. The associated servers form a set, and a random server from this set is nominated as the updated current server. This process continues iteratively, looping back if the current server lacks neighbors and has not reached the maximum server limit until the total servers meet the specified minimum and maximum criteria. The fittest server from the candidate set is then selected, ensuring servers are within the defined range. The procedure repeats while preserving used servers, contributing to a systematic and adaptable service discovery approach in the IoT network, as depicted in Algorithm 1.

```

# Input: Server list, service list
# Output: Selecting best match service
# X: Maximum population capacity
# M: Minimum number of reached servers
# Initialize a population of n servers
InitializePopulation()
# Repeat for a specified number of iterations
for t in range(1, max_iter + 1):
    # Find the best individual in the current population
    X_best = FindBestIndividual()
    # Define a switch probability
     $p = e^{(\max f - \min f) / \max f}$ 
    # Iterate over all nodes in the network
    for i in range(1, N + 1):
        # Check if a random number is less than the switch probability
        if rand() < p:
            # Send an active request for available services
            SendActiveRequest(UR)
            # Receive initially available services
            IS = ReceiveInitiallyAvailableServices()
            # Find the matching score
            matching_score = FindMatchingScore(IS)
            # Check if the minimum number of servers is met
            if NotMetMinimumServers():
                # Delete the population and repeat server selection
                DeletePopulation()
                RepeatServerSelection()
    # Find the current best solution
    current best solution = FindCurrentBestSolution()

```

Algorithm 1. Pseudocode of the proposed method

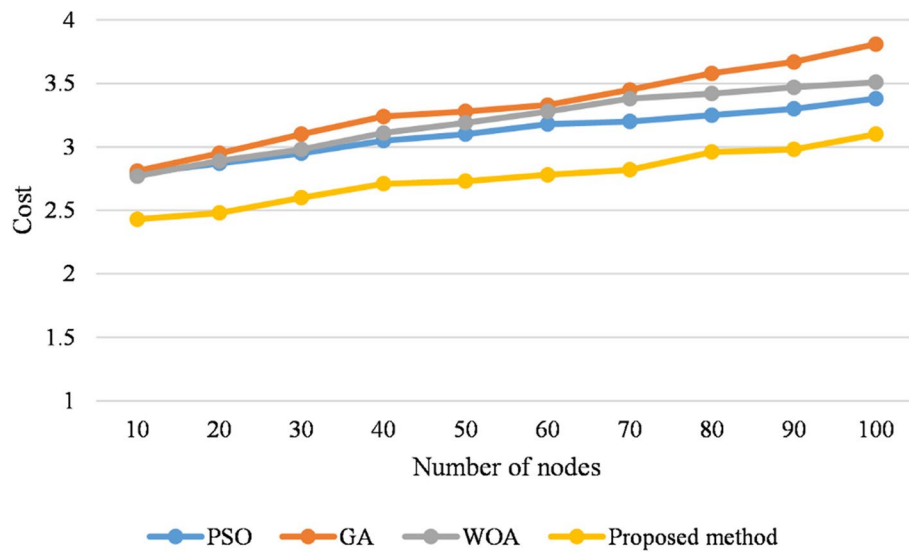


Fig. 5 Cost comparison for 100 tasks

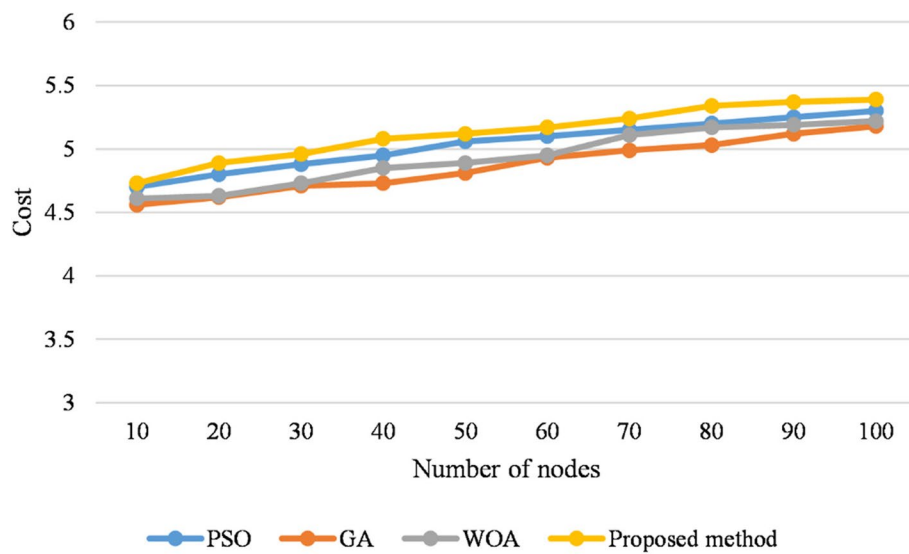


Fig. 6 Cost comparison for 300 tasks

Results and discussion

In this section, the performance of the proposed service discovery technique is compared with that of previous techniques. The experiment outcomes are presented graphically in diagrams and graphs. Test results and comparisons will be presented on time, cost, energy, and convergence diagrams. The suggested technique has been simulated and assessed using the MATLAB simulator. The results of the suggested technique are evaluated and compared with those obtained by PSO, WOA, and GA algorithms.

Figures 5, 6, and 7 illustrate the costs associated with 100 nodes with different numbers of tasks. It is clear that the cost increases in proportion to the number of tasks. However, the percentage increment obtained from the first and third tests, i.e., with

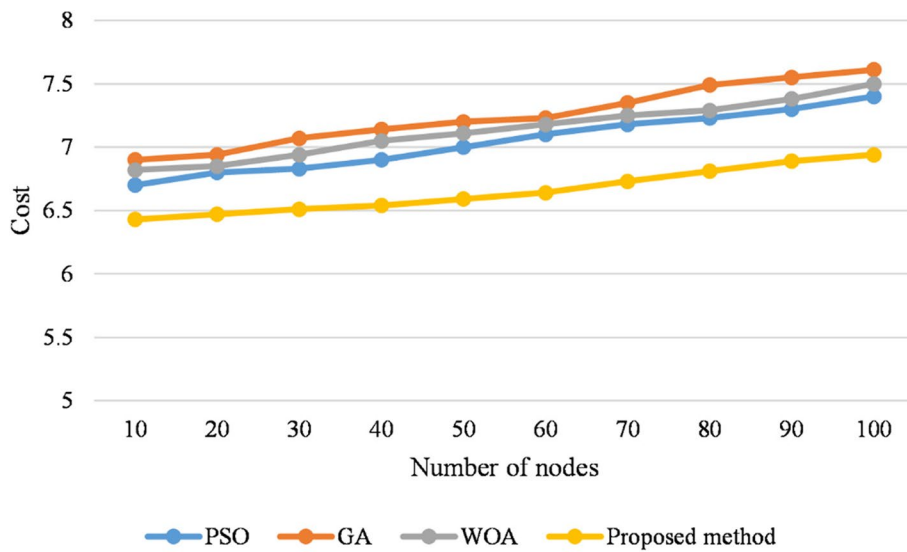


Fig. 7 Cost comparison for 500 tasks

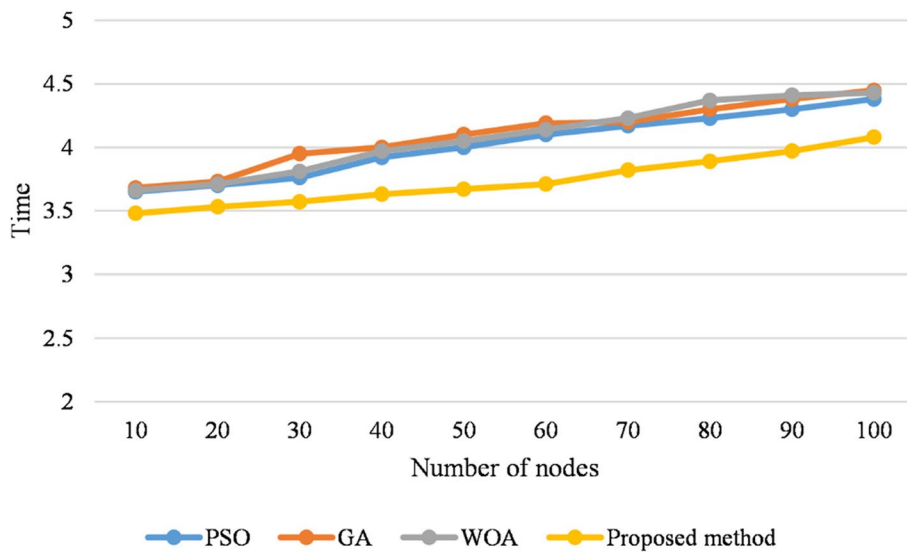


Fig. 8 Time comparison for 100 tasks

100 and 500 tasks, was greater than that obtained from the other algorithms. In the second experiment involving 300 tasks, the genetic algorithm proved more effective than the proposed method. Figures 8, 9, and 10 compare methods in terms of data access between 100 nodes with a variety of tasks. It is evident from the simulation results that the time has increased with increasing the number of tasks, but the rate of this increment is higher in the first two experiments, namely with 100 tasks and 300 tasks, than in the other algorithms. Figures 11, 12, and 13 compare the methods in terms of energy consumption. Increasing the number of tasks has resulted in an increase in energy utilization. However, in the first two experiments, with 100 and 300 tasks, the rate of this increment was higher than the other algorithms. Nevertheless,

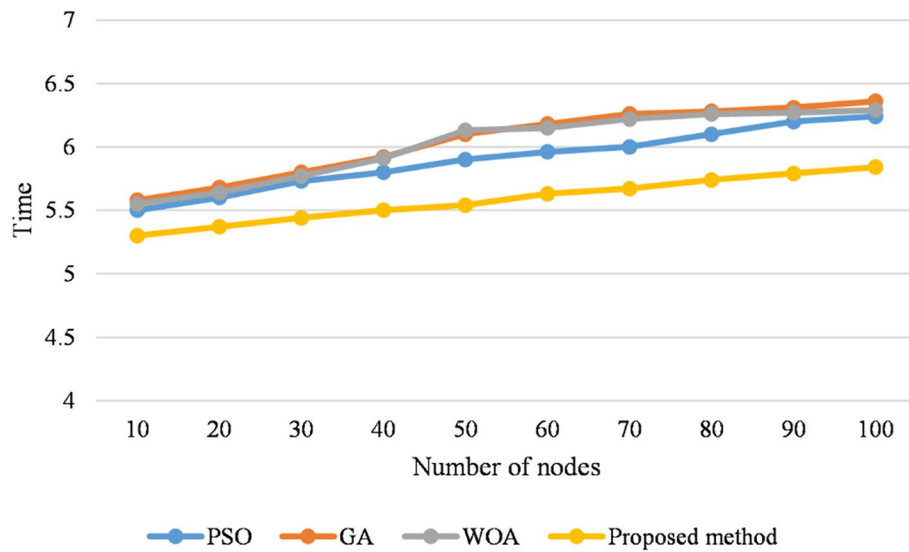


Fig. 9 Time comparison for 300 tasks

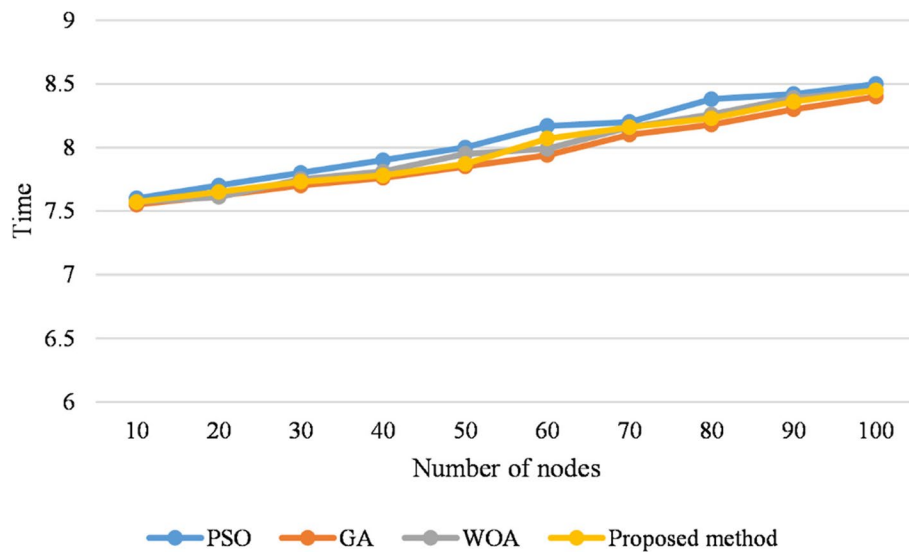


Fig. 10 Time comparison for 500 tasks

the genetic algorithm is superior to the method proposed in this paper in the third experiment with 500 tasks.

Conclusions

The dynamic nature of IoT demands a responsive sensor network infrastructure to handle emerging technologies. The problem of service discovery and selection in dynamic networks is presented as an NP-hard problem in this context. A novel service discovery and selection technique utilizing whale optimization and genetic algorithms is presented in this paper. With our approach, we offer a robust solution to the NP-hard problem of optimal service discovery, reducing the time and cost implications for cloud providers

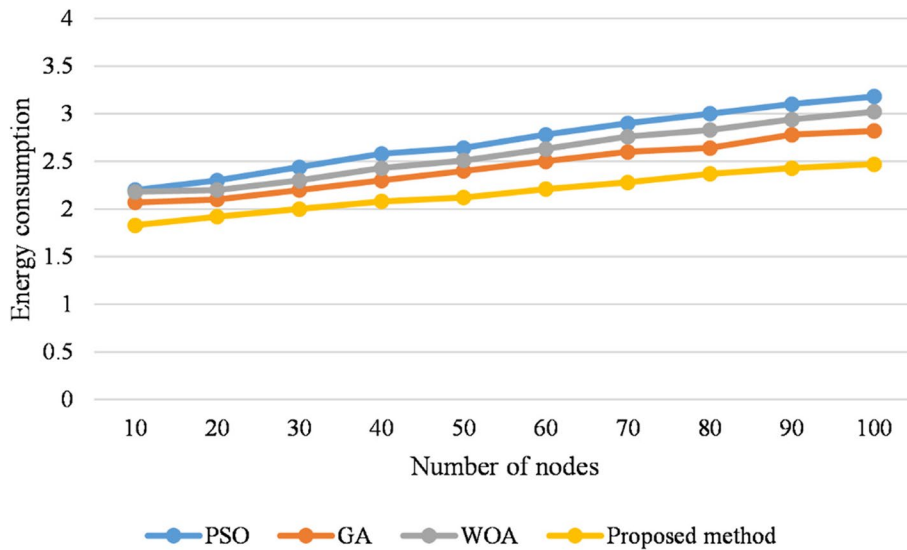


Fig. 11 Energy consumption comparison for 100 tasks

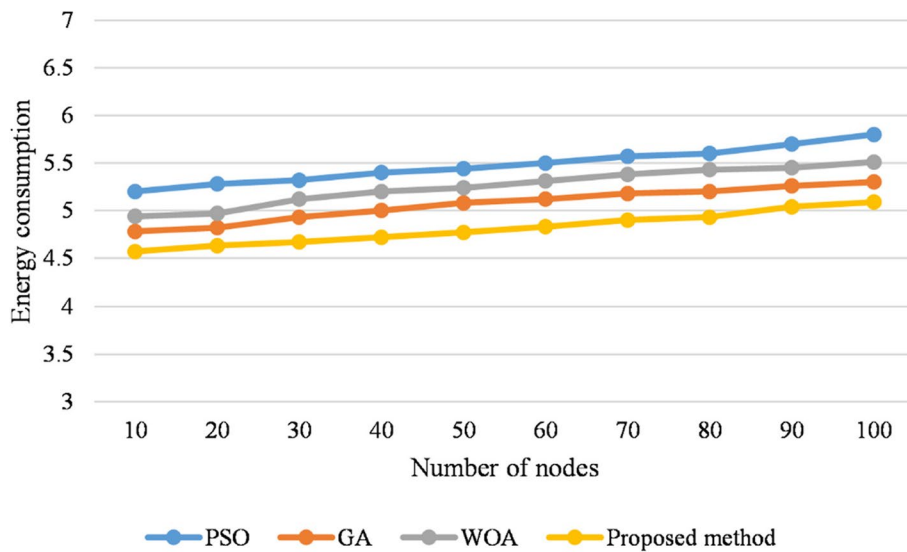


Fig. 12 Energy consumption comparison for 300 tasks

striving to meet QoS requirements. Simulation results confirmed the effectiveness of our algorithm, demonstrating superior performance in comparison to existing baseline approaches. In particular, our method minimizes data access time, optimizes energy usage, and improves cost-effectiveness. Therefore, this research contributes significantly to the field by providing an advanced, efficient, and scalable solution to the challenges of dynamic service discovery in IoT environments. Our work lays the foundation for future developments in enhancing QoS-aware service discovery techniques, ultimately contributing to the evolution of responsive and adaptive IoT infrastructures. While the proposed method demonstrates notable advantages, it is crucial to acknowledge its limitations. The algorithm’s performance may be influenced by specific network

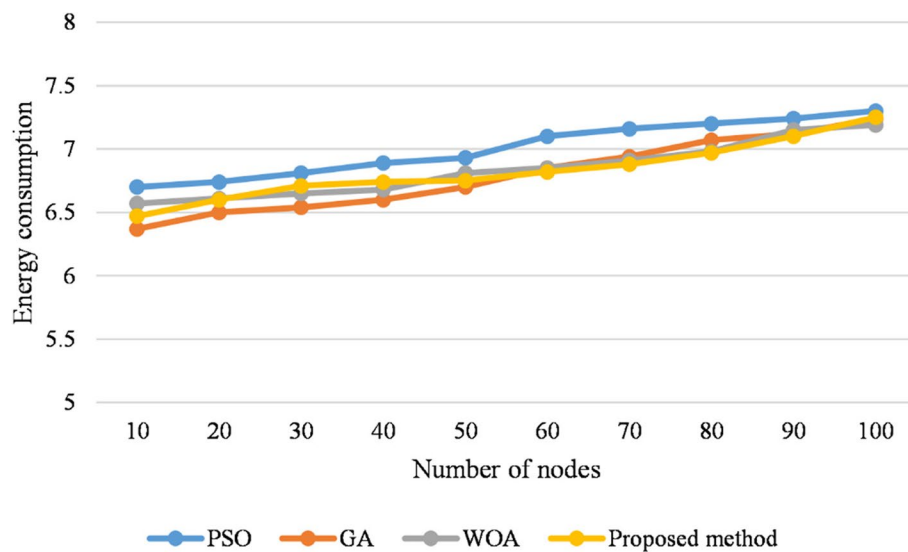


Fig. 13 Energy consumption comparison for 500 tasks

configurations and environmental factors. Future work should explore the adaptability of the approach across diverse IoT scenarios and investigate potential scalability challenges. Additionally, incorporating real-world deployment insights and addressing security considerations will further enhance the robustness of the proposed method. As the IoT landscape continues to evolve, ongoing research endeavors can focus on refining and extending the applicability of the algorithm, ensuring its efficacy in increasingly complex and diverse IoT environments.

Abbreviations

ACO	Ant Colony Optimization
GA	Genetic algorithms
IoT	Internet of Things
IP	Internet Protocol
QoS	Quality of Service
PSO	Particle Swarm Optimization
QoE	Quality of Experience
WSDL	Web Service Definition Language
SR	Service Requester
SP	Service Provider
NDA	Node Data Administrator
SNE	Service Node Evaluator
WOA	Whale Optimization Algorithm
WAG	Weighted Acyclic Graph

Acknowledgements

This work was supported by the project of basic ability improvement of young college teachers in Guangxi: Research on the Construction of a monitoring system for Vocational College students' Learning Ability under the background of digital transformation(No.2023KY1117).

Informed consent

This option is not necessary due to that the data were collected from the references.

Authors' contributions

All authors contributed to the study's conception and design. Data collection, simulation, and analysis were performed by "Xiao Liu and Yun Deng." The first draft of the manuscript was written by "Xiao Liu" and all authors commented on previous versions of the manuscript. All authors have read and approved the manuscript.

Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Availability of data and materials

Data can be shared upon request.

Declarations**Ethics approval and consent to participate**

The observational study conducted on medical staff needs no ethical code. Therefore, the above study was not required to acquire an ethical code.

Competing interests

The authors declare that they have no competing interests.

Received: 6 August 2023 Accepted: 6 December 2023

Published online: 03 January 2024

References

- Pourghebleh B, Navimipour NJ (2017) Data aggregation mechanisms in the Internet of things: A systematic review of the literature and recommendations for future research. *J Netw Comput Appl* 97:23–34
- AbdElaziz M, Al-qaness MA, Dahou A, Ibrahim RA, Abd El-Latif AA (2023) Intrusion detection approach for cloud and IoT environments using deep learning and Capuchin Search Algorithm. *Adv Eng Softw* 176(103402):2023
- Pourghebleh B, Hekmati N, Davoudnia Z, Sadeghi M (2022) A roadmap towards energy-efficient data fusion methods in the Internet of Things. *Concurr Comput Pract Exp* 34:e6959
- Mohseni M, Amirghafouri F and Pourghebleh B (2022) CEDAR: A cluster-based energy-aware data aggregation routing protocol in the internet of things using capuchin search algorithm and fuzzy logic. *Peer Peer Netw Appl* 16:1–21
- Abualigah L, Elaziz MA, Khodadadi N, Forestiero A, Jia H and Gandomi AH (2022) Aquila Optimizer Based PSO Swarm Intelligence for IoT Task Scheduling Application in Cloud Computing, In *Integrating Meta-Heuristics and Machine Learning for Real-World Optimization Problems*: Springer, Berlin/Heidelberg, pp. 481–497
- Kamalov F, Pourghebleh B, Gheisari M, Liu Y, Moussa S (2023) Internet of Medical Things Privacy and Security: Challenges, Solutions, and Future Trends from a New Perspective. *Sustainability* 15(4):3317
- Pourghebleh B, Hayyolalam V, Anvigh AA (2020) Service discovery in the Internet of Things: review of current trends and research challenges. *Wireless Netw* 26(7):5371–5391
- Hayyolalam V, Pourghebleh B, Chehrehzad MR, Pourhajikazem AA (2022) Single-objective service composition methods in cloud manufacturing systems: Recent techniques, classification, and future trends. *Concurr Comput Pract Exp* 34(5):e6698
- Saeidi S, Enjedani S, AlvandiBehineh E, Tehranian K, Jazayerifar S (2023) Factors Affecting Public Transportation Use during Pandemic: An Integrated Approach of Technology Acceptance Model and Theory of Planned Behavior. *Tehnički glasnik* 18:1–12. <https://doi.org/10.31803/tg-20230601145322>
- Nguyen O (2022) HSSCIoT: An Optimal Framework based on Internet of Things-Cloud Computing for Healthcare Services Selection in Smart Hospitals. *Adv Eng Intel Syst* 1(02). <https://doi.org/10.22034/AEIS.2022.343922.1018>
- He P, Almasifar N, Mehbodniya A, Javaheri D, Webber JL (2022) Towards green smart cities using Internet of Things and optimization algorithms: a systematic and bibliometric review. *Sustain Comput Inform Syst* 36:100822. <https://doi.org/10.1016/j.suscom.2022.100822>
- Mahmoudinazlou S, Alizadeh A, Noble J and Eslamdoust S (2023) An improved hybrid ICA-SA metaheuristic for order acceptance and scheduling with time windows and sequence-dependent setup times. *Neural Comput Appl*, pp. 1–19. <https://doi.org/10.1007/s00521-023-09030-w>
- Han C, Fu X (2023) Challenge and opportunity: deep learning-based stock price prediction by using Bi-Directional LSTM Model. *Front Bus Econ Manag* 8(2):51–54
- Sadi M, et al (2022) Special Session: On the Reliability of Conventional and Quantum Neural Network Hardware, In *2022 IEEE 40th VLSI Test Symposium (VTS)*, IEEE, San Diego, pp. 1–12
- Jafari BM, Zhao M, Jafari A (2022) Rumi: An Intelligent Agent Enhancing Learning Management Systems Using Machine Learning Techniques. *J Softw Eng Appl* 15(9):325–343
- Singh R, et al (2022) Analysis of Network Slicing for Management of 5G Networks Using Machine Learning Techniques. *Wir Commun Mob Comput* 2022:10
- Gera T, Singh J, Mehbodniya A, Webber JL, Shabaz M, Thakur D (2021) Dominant feature selection and machine learning-based hybrid approach to analyze android ransomware. *Secur Commun Netw* 2021:1–22
- Aghakhani S, Larijani A, Sadeghi F, Martin D, Shahrakht AA (2023) A Novel hybrid artificial bee colony-based deep convolutional neural network to improve the detection performance of backscatter communication systems. *Electronics* 12(10):2263
- Li Z, Han C and CoitDW (2023) System Reliability Models with Dependent Degradation Processes, In *Advances in Reliability and Maintainability Methods and Engineering Applications: Essays in Honor of Professor Hong-Zhong Huang on his 60th Birthday*: Springer, Switzerland, pp. 475–497
- Shen G, Zeng W, Han C, Liu P and Zhang Y (2017) Determination of the average maintenance time of CNC machine tools based on type II failure correlation. *Eksplotacja i Niezawodność*, 19(4):604–614
- Webber J, Mehbodniya A, Hou Y, Yano K, and Kumagai T (2017) Study on idle slot availability prediction for WLAN using a probabilistic neural network," In *2017 23rd Asia-Pacific Conference on Communications (APCC)*, IEEE, Perth, pp. 1–6
- Nwe NHW, Bao J-M, Gang C (2014) Flexible user-centric service selection algorithm for internet of things services. *J China Univ Posts Telecommunications* 21:64–70

23. Jia B, Li W, and Zhou T (2017) A centralized service discovery algorithm via multi-stage semantic service matching in internet of things. In 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), vol. 1: IEEE, Guangzhou, pp. 422–427
24. Osamy W, Khedr AM, Salim A (2019) ADSDA: adaptive distributed service discovery algorithm for internet of things based mobile wireless sensor networks. *IEEE Sens J* 19(22):10869–10880
25. Pattar S et al. (2019) Progressive search algorithm for service discovery in an IoT ecosystem, In 2019 international conference on Internet of Things (iThings) and IEEE green computing and communications (GreenCom) and IEEE cyber, physical and social computing (CPSCom) and IEEE smart data (SmartData), IEEE, Atlanta, pp. 1041–1048
26. BensalahAzizou Z, Boudries A, Amad M (2020) Decentralized service discovery and localization in Internet of Things applications based on ant colony algorithm. *Int J Comput Digit Syst* 9(5):941–950
27. Jia B, Hao L, Zhang C, Huang B (2021) A privacy-sensitive service selection method based on artificial fish swarm algorithm in the internet of things. *Mob Netw Appl* 26:1523–1531
28. Zhu A, Ma M, Guo S, Yang Y (2022) Adaptive access selection algorithm for multi-service in 5G HETEROGENEOUS Internet of Things. *IEEE Trans Netw Sci Eng* 9(3):1630–1644

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

Submit your next manuscript at ▶ [springeropen.com](https://www.springeropen.com)
