

A Proposed Study to Explore Mechanisms of Resource Discovery and Selection for Intelligent IOT Systems

Dr. Abhishek Tarafder

Abstract

The Internet of Things (IoT) connects computers built into ordinary things to make smart surroundings. In this kind of environment, a user-defined job may be done by combining services from several physical devices. In every Internet of anything (IoT) system, resource discovery is a crucial first step. Resources might be anything like objects, devices, sensors, or data. This research introduces an Adaptive Multi-Hop Clustering (AMHC) method aimed at improving resource discovery processes inside mobile IoT settings. Using cluster heads and gateways, the method builds a strong and fault-tolerant backbone architecture that makes communication easy and resource management scalable. The AMHC method uses a multi-criteria cluster-head election technique that takes into account node energy, connection degree, and path length. This keeps the network stable while spreading the load evenly. AMHC improves cluster stability, finds the best size for clusters, lowers control overhead, and speeds up resource discovery, according to simulation results.

Keywords: Resource discovery, Network, Clusterheads, Topology, Nodes

I. INTRODUCTION

A relatively new technology, the Internet of Things (IoT) allows for the smooth connectivity of smart devices that provide several services, including smart metering, smart traffic, and city monitoring. The significance of the Internet of Things era will be underscored by the exponential growth in the number of IoT devices, including smart appliances, smartphones, security cameras, monitoring sensors, and actuators, which is projected to reach 26-50 billion by 2020. Consequently, it is anticipated that a great deal of focus will be given to ensuring that all Internet of Things devices are highly compatible with one another and can function autonomously under all conditions.

A versatile framework for developing Internet of Things (IoT) applications, the service-oriented computing (SoC)-based IoT vision allows users to construct and reuse on-demand atomic services delivered by IoT devices to complete complicated tasks. To tackle such a daunting endeavour, composite services are created by combining atomic services, which may not separately offer the needed functionality, to produce added-value offerings for IoT applications. System on a chip (SoC) allows for the rapid, inexpensive, and scalable creation and launch of Internet of Things (IoT) applications.

The use of in-depth knowledge is necessary for the discovery of resources. Words on a page, numbers in a table, pictures, files, or packets of measurements transmitted by a computer are all examples of data. Both exponential and linear growth may be used to describe data development, with exponential growth indicating a more dramatic increase. Data cannot be converted from its raw form into information by storing it on a few devices or processing it with a single tool once it has been acquired, prepared, evaluated, and shown functionally. Data that is difficult to store, handle, and analyse using traditional data storage and analytics technologies because to its size, speed, or complexity.

In data analysis, cleaning, modifying, and altering data are common routines. Indexing optimisation, discovery, and ranking solutions are the other two primary focuses of IoT data search. This is closely connected to resource discovery and describes the steps taken, either automatically or by humans, to crawl across IoT settings in search of resources. To make their resources easily accessible, indexing involves categorising the data from the Internet of Things (IoT), which includes retrieving and searching for the data. In order to locate the answer to a query, discovery entails searching for, locating, and investigating data tools according to the primary search attributes. Ranking involves ranking Internet of Things (IoT) resources according to a wide variety of functional and non-functional criteria. In this study, we only look at resources related to the Internet of Things.

In the meantime, resource discovery in mobile IoT networks can be categorised as follows: Time Synchronised Protocols Approach, Deterministic Approach, Colocation-Based Approaches, Fully Distributed Opportunistic Approach, and Context-Aware Resource Discovery Approach.

Society, Industry, and the Environment are the three primary categories into which today's Internet of Things and Industrial Internet of Things technologies fall. Different types of searches are used in different sectors of the Internet of Things (IoT). These types of searches include event-based, location-based, time-related, content-based, spatiotemporal, context-based, real-time, and user-interactive searches. All things considered, the only

ways for search engines to sort results are according to the Text-Based Approach, the Metadata-Based Approach, and the Ontology-Based Approach.

II. REVIEW OF LITERATURE

Tamizan, Mohd & Jamal, Azrul Amri. (2021) This research explores many aspects that lend credence to a knowledge-based approach to the NS-3 simulator environment for the discovery of high-volume IoT resources. Previous studies have already established the idea, particularly using the generally recognised principles of the Q-Learning discovery model. Samples taken from NS-3 simulator emulations of tested data are used to validate the model. The data was modelled and analysed using NS-3 proper simulation based on modules like checkpoint and restore. Using Distributed Multi-Threaded Checkpointing (DMTCP) on a single system and Message Passing Interface (MPI) under distributed machines to speed up the NS-3 model initialisation and execution was the fundamental notion for feasible checkpointing in the NS-3 processes. The Q-learning algorithm approach provides a potential answer for tackling developing IoT settings and setups, and it is the chosen model to be utilised in this investigation. Although Q-learning has shown to be an effective approach for exploring IoT nodes, there are previously known and simplified context-based difficulties such as managing dedicated servers, acquiring data from IoT objects, and meeting the particular requirements of applications. Results show that Q-Learning model improvement is valid for high-volume IoT resource discovery applications. By expanding our understanding of how to conceptualise and validate knowledge-based methodologies grounded on established practices, this study will aid in the creation of new models.

Elmahi, Mohamed et al., (2020) The Internet of Things (IoT) is extensively used in applications that demand instantaneous reaction and interaction. Any Internet of Things (IoT) system must first perform resource discovery, which might take the form of data, services, objects, devices, things, or sensors. The literature presents a number of resource discovery approaches that aim to find certain kinds of resources. In this study, we analyse and categorise current methods for discovering resources in the Internet of Things (IoT) based on their discovery strategy. For objects, things, devices, and services, the article zeroes in on discovery methods. We propose a set of attributes to be utilised as discovery technique assessment criteria and link these attributes to types of discovery strategies.

Tamizan, Mohd et al., (2019) There are a number of topics related to resource discovery in the Internet of Things (IoT) that are still under investigation and verification. When it comes to resource discovery in a high-volume IoT setting, this research compares and contrasts with systematic literature and solution perspectives. The objective is to provide a transparent evaluation of potential resource optimisation and improvement strategies for various IoT systems. After sorting a total of nineteen publications into appropriate categories, we summarised their contents in three tables after conducting a comprehensive literature analysis and comparison of all solutions and literature pertaining to the discovery of IoT resources. We may summarise the key topics of study, IoT platforms, simulation tools, suggested solutions, and categories of IoT resource discovery based on the results. Ultimately, we aim for this evaluation to offer insightful results for all IoT sources and suggest suitable methodologies or procedures to guarantee the future effectiveness of IoT resource discovery. Topics covered include resource discovery, LoRa, the Internet of Things (IoT), the Industrial Internet of Things (IIoT), and high-volume IIoT.

Ullah, Israr & Kim, DoHyeun. (2018) Internet of Things (IoT) applications across several industries have recently seen massive expansion. In the not-too-distant future, it is anticipated that billions of gadgets will be linked to the Internet. In order to construct applications that are based on the Internet of Things, the first step is to create virtual objects. These objects will abstract the attributes of physical devices, allowing you to virtualise them. These virtual objects are then assembled to form various services that find usage in a wide range of applications. Resources offer an efficient means of expressing virtual things in the cyber realm. Even in the absence of actual hardware, in the form of "soft sensors," it is possible to produce physical data or resources that match to a device's logical state. In this article, we offer an OCF IoTivity-based direct discovery technique for managing IoT resources.

Chen, Xu et al., (2018) We provide a novel approach to resource-efficient edge computing in this paper for the next intelligent Internet of Things (IoT) applications, including smart homes, e-health, precision agriculture, and flying ad hoc networks. To ensure that smart Internet of Things (IoT) devices can handle computationally complex tasks, we design an efficient edge computing architecture that distributes them over adjacent devices, helpers, and the edge cloud. We take a different approach than previous research on mobile computation offloading by looking at resource efficiency from a new angle. Based on this, we develop a computation offloading mechanism that minimises the edge resource occupancy of intelligent IoT devices while still meeting their QoS requirements. This mechanism consists of a delay-aware task graph partition algorithm and an optimal virtual machine selection method. The suggested resource-efficient edge computing technique is proven to be effective and performs better according to the performance evaluation.

Perera, Charith & Vasilakos, Athanasios. (2016) With the help of IoT middleware platforms, users of the

sensing as a service model may get the data they need without being privy to the specifics of the sensors and data processing components that make up the IoT. Nevertheless, data consumers face a substantial hurdle when trying to get data and configure an IoT middleware platform; this needs a combination of technical know-how and domain experience. In this work, we present CASCOS, a knowledge-driven method to simplifying the configuration of IoT middleware systems. This will allow data consumers, especially non-technical individuals, to quickly access the data they need. This article shows how the resources of the Internet of Things (IoT) may be semantically represented in a way that allows them to be utilised to build service workflows. The present study is advanced by this automated method of IoT resource creation based on semantic information. We show that our method works by implementing a prototype on an Internet of Things (IoT) middleware called Global Sensor Networks (GSN), but our concept is adaptable to any other platform for IoT middleware.

Chikkamannur, Ajeet & cp, Vandana. (2016) The integration of many smart devices is driving the rapid development of the Internet of Things (IoT) ecosystem. The Internet of Things (IoT) computing paradigm differs from the conventional system in which consumers have complete visibility into the applications and services running on their individual devices. People in an open public environment may be unaware of the resources, services, and capabilities that are accessible to them in the IoT. Therefore, robust human-thing interaction is essential in the IoT. In order to configure and maintain deployed objects and integrate new devices with little human interaction, automatic resource discovery is the key capability. This article discusses the existing state of resource discovery and service technologies in the Internet of Things (IoT), including its benefits and drawbacks. Resource discovery and service integration present unique issues in the IoT ecosystem, and this article examines all of them in detail. Future research and work standardisation can benefit from the supplied information.

III. CHALLENGES IN RESOURCE DISCOVERY

Following is a description of the many difficulties associated with resource discovery and selection:

Heterogeneity in data

An increasing number of sensors, actuators, and mobile devices are now a part of our everyday life, thanks to the expansion of the Internet of Things. The production of massive amounts of data makes it all the more critical to uncover the meaning concealed within this deluge of information. Machines face difficulties in processing and understanding data produced by multi-modal sensors or devices due to the data's vast variations in format, domain, and kind. The inability of data collected by various sensors or devices to communicate with one another is another major obstacle brought about by the current state of interoperability. It is difficult to mine and uncover the concealed information, even though almost 45 percent of data produced on the Internet can be analyzed. The variety of data also makes it harder to communicate information that spans domains.

Semantic interoperability and data management

Data that has to be turned into knowledge is a huge concern in the realm of the Internet of Things (IoT). Standardized data formats, semantic models, and metadata descriptions allow for this change to take place.

Self-organization capabilities

In order for smart things on the IoT to have autonomic capabilities, they must be infused with intelligence. This will allow them to react appropriately based on decisions they make in response to certain situations. Meanwhile, the idea is that smart things may self-organise into temporary ad hoc networks, which will allow them to share data and work together on projects.

Scalability

Naming and addressing, data transfer and networking, information and knowledge management, service delivery and administration, and scalability issues emerge at different levels as the number of connected entities on the IoT increases.

Security and Privacy

Due to the decentralised and virtual nature of the Internet of Things (IoT), the connection between resources is severely limited, making them vulnerable to assaults and malfunctions as they are added dynamically. This means that while designing architectures and methodologies for discovery solutions, security should be considered a crucial attribute at the system level.

IV. PROPOSED ALGORITHM

Adaptive MultiHop Clustering (AMHC), which produces an effective MANET backbone architecture, makes Resource Discovery (RD) tasks simpler. Rapid reconstruction of the backbone architecture is performed whenever there is a change in the network topology as a result of node relocation or resource scarcity. The fundamental components of the backbone are the headers and gateways of every MANET site. Amhc promotes re-electing clusterheads in following rounds to reduce transition overheads when old clusterheads transfer all resource information to new clusterheads. It builds a backbone with many channels linking close clusterheads

using all the gateways, which improves fault tolerance and decreases backbone congestion. Amhc aims to accomplish both load-balancing and stability, despite the fact that these are competing goals. Very stable clusterheads may carry more weight and drain their energy resources more quickly. When transferring cluster resource tables from one head to another, insufficient stability causes performance degradation and battery resource consumption due to the needless communication it requires.

As the AMHC algorithm iteratively selects a node with the highest weight based on degree of connectivity, path length, and current available energy, the process continues over d rounds of data exchange. A node with the highest weight in its immediate surroundings is called a "clusterhead" in this context.

As a first step in maintaining a stable neighbourhood, each node determines its own weight and periodically broadcasts it to its one-hop neighbours via beacons. In the data, you may find details like the current energy available and the identifiers of the nodes. A "node id" can be any legitimate identification deriving from any layer; this includes MAC addresses and IP addresses, among others. Node id isn't strictly necessary for AMHC to function, however it does help when using the hashing technique to index asset databases. Clusterheads that do not yet have an identifier are first assigned a made-up number. When all of the nodes in its immediate vicinity have finished transmitting, the node that is now serving as clusterhead will choose one by tallying up the weight values received and the weight value currently held by the winning node. After d rounds of this process, the node having the most weight at the end of storage is chosen as the clusterhead.

When another clusterhead passes by, the first one can break apart from its cluster. As seen in Figure 1, clusterhead A grows by taking up three other clusters' real estate, whereas cluster E is untouched. Although A completely consumes Cluster B, B and C are only partially consumed. Only in cluster C does the problem of untangling the leader become apparent. In order to resolve the issue, AMCH grows cluster A at the end of the d -round until it engulfs all of cluster C. This is why certain nodes in the cluster might not be able to go more than two degrees away from the leader. After a certain number of successive beacons, the discovery of a neighbour who isn't around due to power outages or mobility problems happens. A node whose neighbourhood varies over time and whose mobility is entirely random is considered in this model. Since nodes may not always be aware of what's happening in their near neighbourhood, it is vital to update the formed clusters and their heads often to keep the network topology accurate.

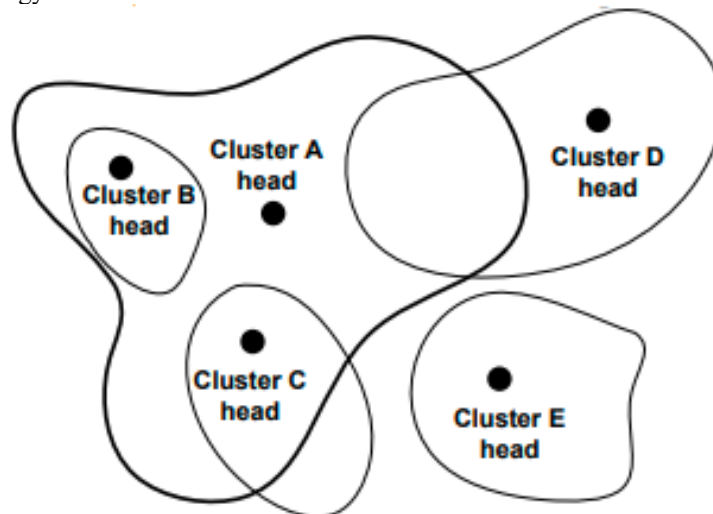


Figure 1: Different types clustering takeover during the creation of multi-hop clusters.

A few simple variables and a vector to hold the one-hop neighbour list are all that's required for each node. Notable instances of such messages include HELLO, which is sent whenever a node wishes to rejoin the network (e.g., after being separated and subsequently wanting to rejoin or being brought online due to resource issues), ACK-HELLO, which is sent whenever a node receives one or more HELLO messages within the neighbourhood discovery time interval, and CH-ELECTION, which is sent periodically throughout the d rounds of the election process to publish the identity and weight of the elected clusterhead. In the first round, all of the nodes utilise their own algorithms to calculate their weight and act as clusterheads. It may register itself as a member of a cluster using the REGSITER command and ask other nodes to join a nearby cluster using the JOIN command. With a far lower message volume, the clustering heuristic achieves an $O(d)$ communication round complexity, which is lower than that of competing techniques. Among the many advantages offered by AMHC are distributed responsibility, small storage requirements, a fault-tolerant backbone, an equilibrium between inter- and intra-cluster communication, and a minimal number of clusters that depends on the parameter d .

V. EXPERIMENTAL SETUP

When testing the AMHC algorithm in a simulation, the following parameters are used: network size (250 m x 250 m), number of nodes (100 to 1000), node speed (1.5 to 50 km/h), transmission range (30 to 300 m), simulation time (15 minutes without breaks), and a random mobility model. Using the Max-Min Clustering Algorithm, Maximum Connectivity Clustering (MCC), and the Lowest-ID Algorithm as benchmarks, we assessed AMHC's efficiency. In order to simulate ad hoc networks with varying node densities, we conceptualised several systems with 100, 200, 400, and 600 nodes, respectively.

We say that two nodes have a wireless link when they are physically near enough to each other to create a communication channel. We tested the system in a virtual environment by simulating the communication ranges of nodes with 30, 40, and 50 length units. On top of that, the clustering parameter d was initially set to 2 and then to 3 for all of the previously described simulation combinations. It took 2000 seconds for the simulation to complete, with a network sample every 2 seconds. The most crucial statistics for a simulation are the number of clusters, the size of clusters, and the duration of clusterheads. With these figures, we were able to evaluate the efficacy of the proposed strategy. A single hop is all it takes for algorithms like MCC and Lowest-ID to create clusters. Before we can make an accurate comparison between these approaches and the AMHC, we run d -closure on the connection topology. This changes the graph such that in the current topology, two nodes are considered one-hop neighbours if they were previously no more than d hops apart.

VI. RESULTS AND DISCUSSION

The outcomes of AMHC are shown in Figure 2 for MANETs with different transmission ranges and node counts. The mobile nodes keep moving at a speed of 20 meters per second throughout the whole network simulation. We see a rather high average number of clusters when the transmission range is moderate (10–20 m). With an increase in transmission range, the number of nodes that are able to hear one another increases. Cluster sizes grow in proportion to the decrease in the average number of clusters produced. Since mobile nodes have a large transmission range, they often remain within range of their neighbours. Consequently, clusters become less dynamic and cluster head changes occur less frequently.

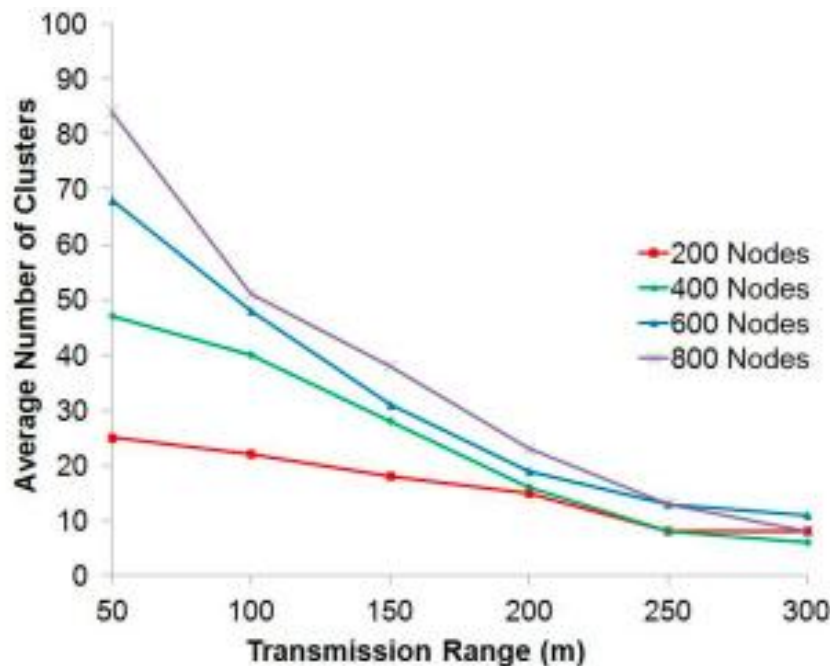


Figure 2: Impact of network size and transmission range on average number of clusters

Figures 3 and 4 show that out of the four algorithms tested, AMHC yields the most cluster heads on average. Large cluster sizes, high levels of communication inside clusters, and low levels of communication between clusters are all signs of a low cluster density. Greater clusters utilise their energy (and other resources) more quickly due to the increased amount of work they are accountable for. For larger numbers of cluster heads, the converse was true: low levels of intra-cluster communication and high levels of inter-cluster communication were associated with smaller cluster sizes. When there are many clusters, the resources of individual nodes are quickly depleted due to factors such as sluggish overall performance, delays in resource requests, and excessive traffic across gateways.

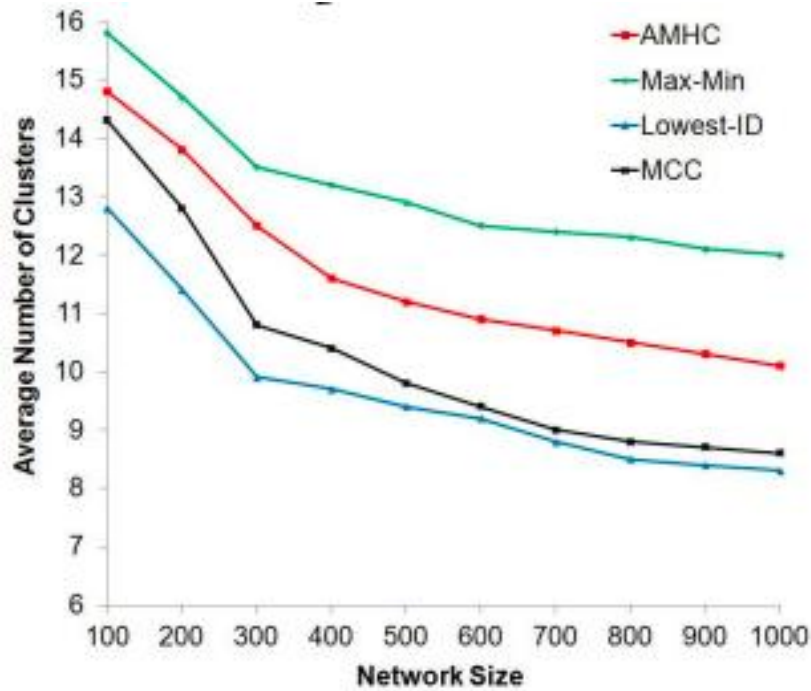


Figure 3: Average number of clusters

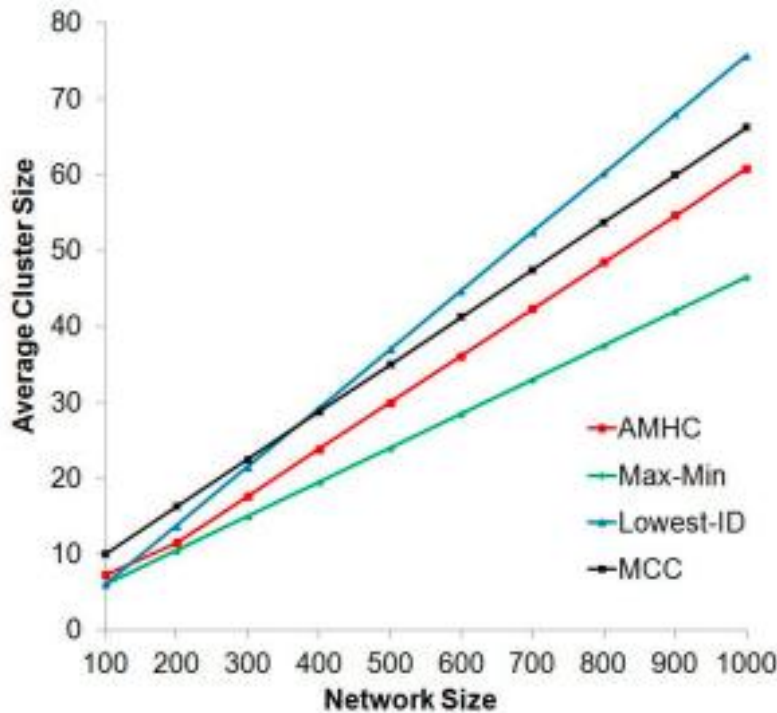


Figure 4: Average cluster size

Clustering algorithms are crucial for adapting to changing network topologies. There needs to be little reaction to these changes in order to keep stability high while reducing overhead. Figure 5 shows that out of the four clustering methods, AMHC has the longest average cluster head length. AMHC-generated, extremely stable clusters are capable of performing RD tasks for long durations. Based on the results of the simulations, the AMHC algorithm provides the best results for overall cluster leader election features.

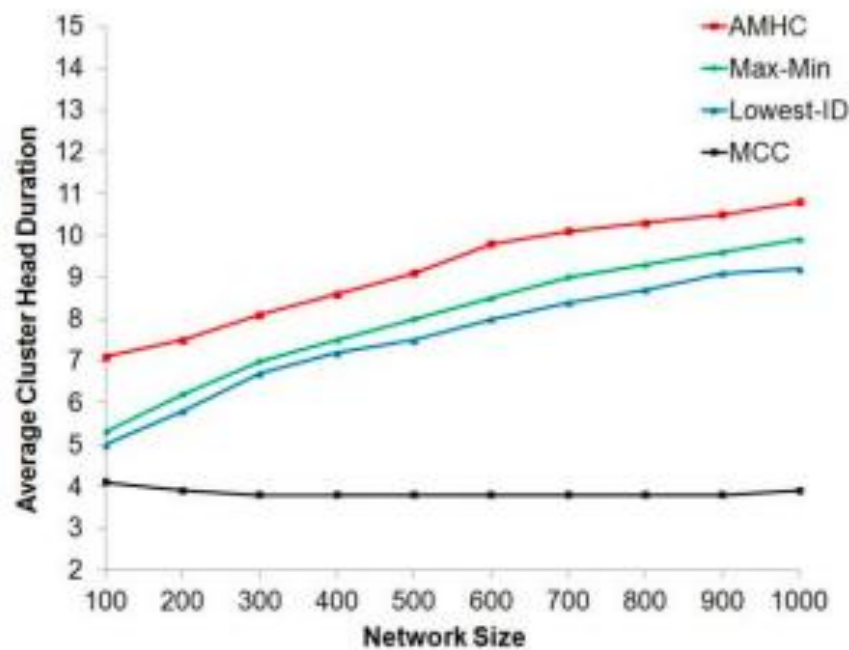


Figure 5: Average clusterhead duration

VII. CONCLUSION

To provide a robust and reliable backbone architecture that facilitates simple communication and control of resources throughout the whole network, the suggested approach makes good use of clusterheads and gateways. Node energy, connectedness, and path length are only a few of the clusterhead election criteria used by the AMHC technique, which aims to strike a compromise between load distribution and network stability. When compared to traditional clustering methods such as Lowest-ID, MCC, and Max-Min, the simulation-based evaluation reveals that AMHC outperforms them in terms of communication efficiency, clusterhead duration, and cluster stability. The approach is effective even when the network design undergoes modifications, as nodes relocate and transmission ranges fluctuate. It optimizes the efficiency of resource discovery tasks and costs less as well. Since AMHC is both scalable and lowers network congestion, it is an excellent option for big Internet of Things (IoT) installations. It has multi-hop clustering and a d-round election process.

REFERENCES:

- [1] V. cp, S. Basha, M. Madijagan, S. Jadhav, M. Matheen, and L. Maguluri, "IoT resource discovery based on multi-faceted attribute enriched CoAP: Smart office seating discovery," *Wireless Personal Communications*, vol. 146, no. 1, pp. 39–56, 2024.
- [2] K. Chakraborty, D. Kapila, S. Kumar, S. Nazeer, and A. Singh, "Intelligent machine learning based Internet of Things (IoT) resource allocation," *Engineering Proceedings*, vol. 59, no. 1, pp. 1–7, 2023.
- [3] M. Tamizan and A. A. Jamal, "Determination of suitable resource discovery tool and methodology for high-volume Internet of Things (IoT)," *Journal of Physics: Conference Series*, vol. 1874, no. 1, pp. 1–11, 2021.
- [4] M. Samadi, "A knowledge and intelligent-based strategy for resource discovery on IaaS cloud systems," *International Journal of Grid and Utility Computing*, vol. 12, no. 2, pp. 205–221, 2021.
- [5] M. Elmahi, N. Osman, and H. Hamza, "Resource discovery classification for Internet of Things: A survey," *International Journal of Digital Information and Wireless Communications*, vol. 10, no. 3, pp. 35–50, 2020.
- [6] M. Tamizan, A. Bakar, A. A. Jamal, and E. N. Madi, "Resource discovery in high-volume Internet of Things: Systematic research," *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, no. 12S2, pp. 437–444, 2019.
- [7] Ullah and D. Kim, "IoT resource management using direct discovery mechanism in OCF framework," *International Journal of Grid and Distributed Computing*, vol. 11, no. 5, pp. 1–10, 2018.
- [8] X. Chen, Q. Shi, L. Yang, and J. Xu, "ThriftyEdge: Resource-efficient edge computing for intelligent IoT applications," *IEEE Network*, vol. 32, no. 1, pp. 61–65, 2018.
- [9] A. Jamal and W. J. Teahan, "Alpha multipliers breadth-first search technique for resource discovery in unstructured peer-to-peer networks," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 7, no. 4, pp. 1403–1412, 2017.
- [10] Perera and A. Vasilakos, "A knowledge-based resource discovery for Internet of Things," *Knowledge-Based Systems*, vol. 109, no. 1, pp. 1–16, 2016.
- [11] Chikkamannur and V. cp, "Study of resource discovery trends in Internet of Things (IoT)," *International Journal of Advanced Networking and Applications*, vol. 8, no. 3, pp. 3084–3089, 2016.
- [12] L. Nunes, J. Estrella, C. Perera, S. Reiff-Marganiec, and A. Delbem, "Multi-criteria IoT resource discovery: A comparative analysis," *Software: Practice and Experience*, vol. 47, no. 10, pp. 1–16, 2016.
- [13] N. J. Navimipour and F. S. Milani, "A comprehensive study of the resource discovery techniques in peer-to-peer networks," *Peer-to-Peer Networking and Applications*, vol. 8, no. 1, pp. 474–492, 2014.
- [14] N. J. Navimipour, A. Rahmani, A. Navin, and M. Hosseinzadeh, "Resource discovery mechanisms in grid systems: A survey," *Journal of Network and Computer Applications*, vol. 41, no. 1, pp. 389–410, 2014.

- [15] F. Tao, Y. Zuo, L. Xu, and L. Zhang, "IoT-based intelligent perception and access of manufacturing resource toward cloud manufacturing," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 1547–1557, 2014.
- [16] J. Albrecht, D. Oppenheimer, A. Vahdat, and D. A. Patterson, "Design and implementation trade-offs for wide-area resource discovery," *ACM Transactions on Internet Technology*, vol. 8, no. 4, pp. 1–44, 2008.