Indian Journal of Pure & Applied Physics
Vol. 62, November 2024, pp. 996-1003
DOI: 10.56042/ijpap.v62i11.12403

Optimizing Master-Slave Communication in Collaborative Robotics Using IoT-Enabled Robot Interaction

Ayush Kumar Agrawal & Jayendra Kumar*
Department of Electronics and Communication Engineering, National Institute of Technology Jamshedpur, India

Received 11 July 2024; accepted 10 October 2024

Swarm robotics involves coordinating the behaviour of multiple robots to perform complex tasks collaboratively. Our study explores using low-cost NodeMCU ESP8266 devices to establish master-slave communication in swarm robotics utilizing the Internet of Things (IoT), focusing on different network topologies: star, mesh, and ring. Unlike existing research that uses more expensive or less accessible technologies, our approach demonstrates that these affordable devices can effectively manage communication and coordination in swarm systems. Our experiments reveal that the star topology is the most efficient for master-slave communication, allowing for aggregation, pattern formation, and coordinated motion tasks. This research highlights the potential of using low- cost devices for efficient communication in swarm robotics, offering significant implications for various industries.

Keywords: Swarm robotics; NodeMCU ESP8266; Master-slave communication; Collaborative robots; Network topologies

1 Introduction

The Internet of Things (IoT) is rapidly transforming how we interact with technology by connecting various devices, sensors, and software over the internet. With its potential to revolutionize industries such as transportation, healthcare, and city planning, the IoT has become a significant topic of interest in computer technology. However, despite the advancements, a key challenge remains in establishing seamless communication and interaction among diverse IoT based robots due to the lack of a unified framework. This paper addresses this gap by proposing solutions to enhance communication within two collaborative or swarm robots.

One of the critical features of the IoT is its ability to employ Artificial Intelligence (AI) techniques such as machine learning (ML) and deep learning (DL), which can enhance the intelligence of the system and improve its ability to perform complex tasks. However, a unified framework is needed to enable seamless communication and interaction among IoT devices. To address this challenge, we propose using NodeMCU ESP8266, a low-cost, compact device with an in-built Wi-Fi module, as a communication module for the IoT. This device can be used in various applications, including wireless sensor networks and swarm robotics, where multiple small devices must work together to achieve a desired outcome.

Swarm robotics is a promising research area involving coordinating many simple robots, or termed collaborative robots, to achieve collective intelligence and behaviour. By applying swarm intelligence principles to collective robotics, we can create a new paradigm for distributed computing and autonomous systems[1,2]. In this paper, we explore using NodeMCU ESP8266 in swarm robotics and propose different network topologies for establishing communication among the robots. Our research has important implications for various industries, including medical, defence, agriculture, and construction. By enabling seamless communication among IoT devices and employing swarm intelligence principles, we can develop new solutions for complex problems and create more efficient, sustainable systems[3].

The clustered dynamic task allocation method inspired by the approach of the Particle Swarm Optimization (PSO) algorithm, which performs entirely distributed task allocation in a swarm robotic system, uses the notion of adaptive speed to do a supervised search of the allocation space. Using a cluster communication topology amongst swarm robots is proposed to optimise fundamental communication and enable effective job sharing for giant robots of swarms. The findings of the cluster-based topology are compared to those of the entire mesh-based topology [1].

*Corresponding author: (E-mail: jkumar.ece@nitjsr.ac.in)

Topologies of intermediate particle connection were also investigated, and the global/local performance gap was considerably more pronounced. To identify big advances from tweaks, a metric of significant improvement is established[4].

Improved butterfly optimisation algorithm version is used for Internet of Things (IoT) based Home Energy Management System (HEMS) and used ZigBee-based wireless technology. It is used to improve the system's efficiency for the cost of energy consumption and level of user satisfaction. It also compared the PSO and BOA algorithms by observing the power consumption of different items before and after optimisation with other algorithms. Similarly, unmanned aerial vehicle (UAV) network systems based on the Internet of Things (IoT) and cloud server are implemented to track the air quality in real-time for intelligent city use. It will detect and record the exact cause of pollutants like Carbon monoxide (CO), Nitrogen dioxide (NO2), Sulphur dioxide (SO2), Volatile Organic Compounds (VOC), Ozone, furans, dioxins etc in the site using a camera and sensor,, respectively and also uploaded the sensor data in the cloud server which can be monitor and alert when it reaches its threshold value[5-8]. With minimum sensor technology to achieve low cost and energy efficient system for seeding using a centralized entity responsible for planning, optimization and supervision, Mobile Agricultural Robot Swarm (MARS) is designed, it uses Cluster-Based Local Updates (CLU) to refill the MARS in case any robot tasks are empty. Also, the Vehicle Routing Problem with Drones (VRPD) is designed to increase demand in e-commerce and logistic supply, assign customers to drone-truck pairs, determine the number of dispatching drone-truck units, and obtain optimal service routes. In contrast, the fixed and travel costs of both vehicles are minimised. It uses the Ant Colony Optimization (ACO) algorithm to solve the different size instances and customer location distribution problems of Vehicle Routing Problem with Drones (VRPD).

The current study on the use of NodeMCU ESP8266 devices for master-slave communication in swarm robotics builds upon established research in communication technologies and swarm intelligence. Clerigues *et al.* (2024) emphasized the necessity of robust multi-hop wireless communications for resilient UAV swarms, highlighting the critical role of efficient data transfer in maintaining system integrity and performance[1]. Similarly, Huang *et al*. (2022) addressed optimization strategies for vehicle routing with drones, underscoring the importance of effective communication networks in enhancing operational efficiency[2]. Our results, which indicate the superior performance of the star topology, are consistent with the findings of Agrawal *et al.* (2021), who explored innovations in robotics and noted the impact of communication strategies on system effectiveness[3]. The cost-effectiveness and applicability of the NodeMCU ESP8266 devices are supported by Bento (2018), who provided a comparative analysis of IoT devices, reinforcing their suitability for various applications[5]. Additionally, Sharkey and Sharkey (2006) offered foundational insights into applying swarm intelligence to collective robotics, corroborating the observed efficiency of the star topology in our study[9]. The work of Navarro and Matía (2013) on swarm robotics further highlights the significance of practical communication topologies in swarm coordination and performance[4]. Liekna and Grundspenkis (2014) reviewed practical applications of swarm robotics, aligning with our findings on the advantages of the star topology in real-world scenarios[6]. Furthermore, Bharti and Agrawal (2020) discussed the communication needs of advanced robotic systems, which supports our selection of cost-effective solutions for improved system performance[7]. Finally, Sivakumar *et al*. (2021) addressed the integration of IoT and wireless sensor networks in surveillance applications, emphasizing the potential of low-cost technologies to enhance functionality across various domains[8].

This paper implements three topologies to communicate among the robot swarm using NodeMCU ESP8266. Topologies which used to communicate among the swarm are, a) Master-Slave, b) Mesh, and c) Ring. NodeMCU ESP8266 can communicate with each other with or without using the internet. To speak without the internet, NodeMCU can use the device's MAC address, whereas communication using the internet requires an intermediate device like a router to communicate. Master-Slave method will used to study in both types, *i.e.* with or without using the internet, which also leads to implementation in IoT. IoT is used to monitor the data over the internet in which updating of sensor data over the Thing Speak cloud server is achieved[9]. But for mesh and ring topologies, only one communication method, *i.e.*, communication without the internet, is used.

2 Communication Modes for Swarm Robot

Swarm robots communicate with each other to perform collective operations, which makes it necessary to study the communication among swarm behaviours. Here, three different ways of communication among the swarm are used and demonstrated, which are a) Master- Slave method, b) Mesh, and c) Ring, where individual bots in the swarm will perform their operation. When the master-slave method is used, it will demonstrate a centralized way of communication among the swarm. In contrast, when mesh or ring topology is used, it will demonstrate decentralised communication among the swarm. For master-slave communication, two ways are used: the first is without using the internet, and the second is using the internet. Communication without using the internet will be done by using the MAC address of the individual device, and communication using the internet will use an intermediate medium like a router to communicate with others. Also, it will use IoT to read and write data over the cloud server[10–12]. For mesh and ring topology, there is no requirement for any intermediate medium to communicate with others; it can only communicate with the help of the MAC address of the devices.

For displaying the intelligence behaviour among the swarm, a microprocessor for computing and processing the data is needed as a communication device that will establish communication and transfer the data. NodeMCU ESP8266 is used as a microprocessor for processing inputs and can interface TCP/IP communication over IEEE 802.11 wifi standard. To implement the master-slave communication among the devices using IoT, a Thing Speak cloud server is used, where the slave can update (write) their data, and the master can update (write) its data and also fetch (read) data of the slave from the Thing Speak cloud server[13,14]. One of the essential characteristics of the swarm is to sense the external environment. For sensing the external environment ultrasonic sensor HC-SR04 is used. Ultrasonic sensor HC-SR04 is an affordable sensor that offers non-contact measurement capabilities from 2 cm to 400 cm with a range accuracy of up to 3 mm. Measurement of distance using ultrasonic sensor starts with Trig pin at high pulse (5V), at least for 10us. It initiates the sensor, transmitting out 8 cycles of ultrasonic burst at a frequency of 40 kHz and waiting for the ultrasonic burst reflected from the obstacle. When the ultrasonic sensor receives ultrasonic from the receiver, it sets the Echo pin to high (5V). Delay/Duration for the period is proportional to distance.

Duration = Width of Echo pulse (in microseconds)
*Distance = Duration*0.034/2*

Below, Fig. 1 shows the interfacing of the individual device using NodeMCU ESP8266 and ultrasonic sensor HC-SR04 and a combination of this same circuit in using three NodeMCU ESP8266 to observe the behaviour of the swarm using the mentioned topology.

3 Communication among Swarm Using Master-Slave Topology

In the master-slave method, one device will act as master and the other two work as slave in the swarm. This type of communication will establish where slaves can share the data with the master for monitoring and instructing slaves if required. Slaves can also observe the master's data but cannot instruct the master in any operation. So, the data transfer between the master and slave is bidirectional, whereas instruction transfer is unidirectional[15-18].

The master-slave method demonstrates the master-slave communication among the swarm with and without the internet. For communication without the Internet, we will use the MAC address of the individual module, whereas, with the help of the Internet, we will use an intermediate medium like a router to establish communication. Also, it will use IoT to read and write data over the cloud server. For

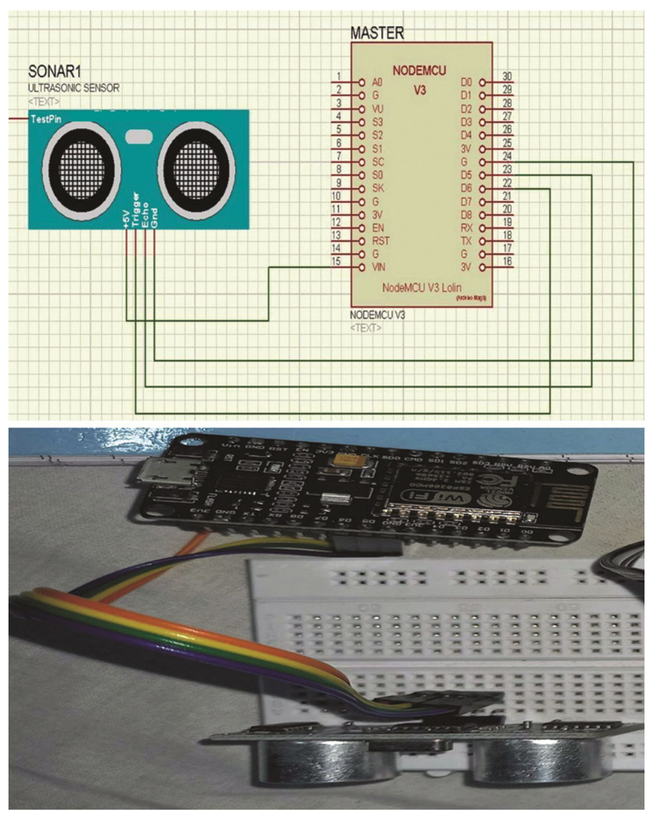

Fig. 1 — Circuit (diagram and connection) of Individual Swarm

observing the master-slave among the robots of the swarm, we have designed and established the connection using three NodeMCU ESP8266 interfaced with the ultrasonic sensor HC-SR04. Each NodeMCU uses IEEE 802.11 wifi standard to develop communication among the swarm. The circuit of interfacing the ultrasonic sensor with master and slaves NodeMCU ESP8266 and communication establishment among the swarm is given in Fig. 2.

3.1 Master Slave Communication among Swarm without Using Internet

In this, master-slave communication among the swarm is established using the MAC address of the individual NodeMCU. The master will communicate with both the slave using IEEE

802.11 wifi standard. It can be said that the master will create a local hotspot for the slave to connect with it. This type of communication among the swarm does not have to depend on the medium of the internet to perform any operation. Here master will have a more significant task to perform. Some of them are: a) perform its operation, b) monitor the incoming data from the slave, c) process the incoming data from the slave, and d) pass appropriate commands to the slave if any fault or abnormality is observed. On the other hand, the slave will perform operations like a) perform its operation and b) monitor the incoming data from the master1[9,20]. The basic block diagram of this type of communication among the swarm is given in Fig. 3. Arduino IDE is used to program every individual NodeMCU ESP8266.

The ESP-NOW protocol is used to establish communication among the different NodeMCUs using the MAC address of the devices. It is a connectionless wifi communication protocol defined by Espressif. ESP-NOW encapsulated application data in a vendor-specific action frame and, without

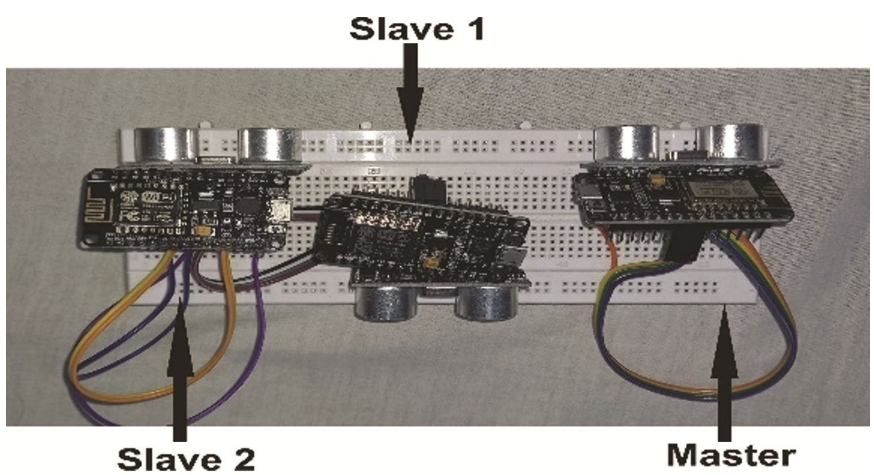

Fig. 2 — Circuit of the Master-Slave among different NodeMCU interfacing with HCSR04

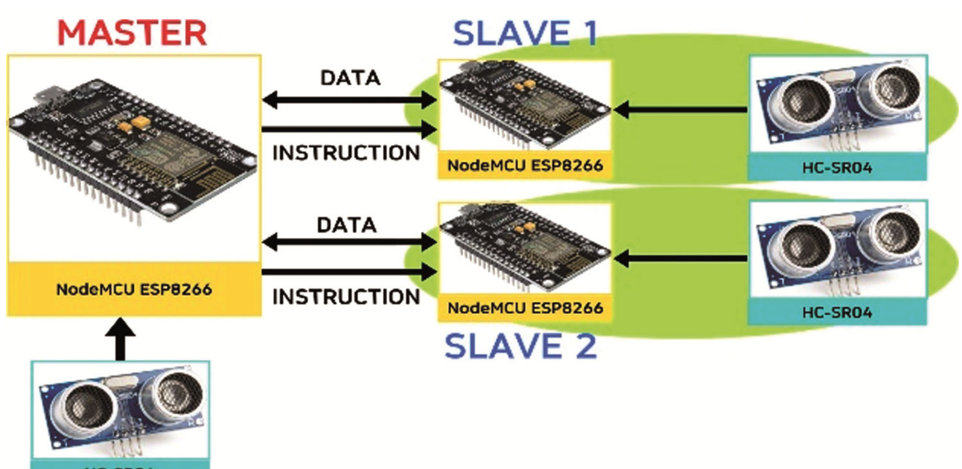

Fig. 3 — Block Diagram of Master-Slave Communication among Swarm without using Internet

connection, transmitted over Wi-Fi from one device to another. CTR protects the action frame for security with the CBC-MAC Protocol (CCMP). ESP-NOW is used in remote control, intelligent lights, sensors, etc. Some of the primary functions which are being used to establish communication transfer of data and receiving of data are as follows:

- *esp_now_set_self_role ()* - Set the role of the NodeMCU device.
- *esp_now_send()*- Sent the data to the receiver MAC address.
- *esp_now_register_recv_cb ()* - Receive data from receiver.

One of the main advantages of using this method is that it does not require network availability, i.e., it can perform an assigned operation ind`ependently of its location. Also, using this swarm method, it can easily monitor the operation of the swarm. Apart from this, there are some disadvantages of using this swarm technique, such as that it can only operate on a small scale and in a small area.

3.2 Communication among Swarm Using Internet (IoT)

This type of communication strategy requires the internet to communicate among the master- slave devices, i.e., form an IoT network for the devices to operate. In this communication technique, the master and slave use NodeMCU ESP8266 and an ultrasonic sensor. Here, both master and slaves can be connected using an intermediate device such as a router to access the internet. For this, the Thing Speak cloud server updates and fetches data. Here, one of the devices is used as a master, and the other two are used as slaves, and both master and slave can write the data to the Thing Speak cloud server[21]. The master will be able to operate, a) its operation, b) monitor the data from the slaves, c) process the incoming data from the slave, d) instruction required for the slave in case of any malfunction and e) write the data over ThinkSpeak cloud server and also read slave data from the cloud server.

The slave will have to operate: a) individual operation of the bot, b) monitor the data from the master, and c) write the sensor's data to the cloud server. In this IoT-based communication network, the master and slave use an application programming interface (API) key to read and write data on a particular channel. We can also monitor the data from any remote location in real-time. With the help of IoT, it can access the data of any sensor in real-time. This technique will overcome the problem where a swarm can only operate on a small scale and in a small area. It can also use a different swarm intelligence algorithm for optimization[22]. This will make the system accessible anywhere. ThingSpeak cloud will be used for the storage of data too. To read and write the data from the cloud server using the API key is shown in Fig. 4. As from above, Fig. 4 displays that storing or accessing the data over TCP/IP must be connected to the internet to establish the communication.

So firstly the device will request internet access. According to the response, the device will use the API key of the particular channel to connect to the web server. Establishing the connection with the web server is one of the essential tasks because only after that will devices be able to write or read the data from the database according to the device's needs. For writing and reading the data over the ThingSpeak server, we use the Arduino IDE ThingSpeak library. It will provide a write and read function for the data using the API key as in Fig. 5 and Fig. 6, respectively.

For writing the data over the cloud, the ThinkSpeak.write Field() function requires a channel ID, field number where it can write the data, data which needs to be written and the API key of the

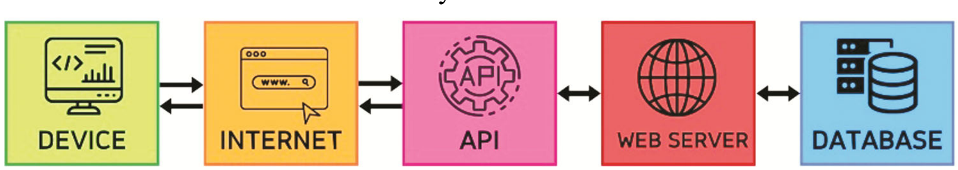

Fig. 4 — Working of API

```
ThinkSpeak.writeField(myChannelNumber, 1, distance, myWriteAPIKey);
```

Fig. 5 —— Write function of the Thing Speak Library

```
ThinkSpeak.readLongField(myChannelNumber, FieldNumber2, myReadAPIKey);
```

Fig. 6 —— Read function of the Thing Speak Library

channel. Similarly, while reading data from the cloud, the ThinkSpeak.read Long Field () function requires channel ID, field number from where it reads the data and the API key of the channel.

4 Communication among Swarm Using Mesh Topology

The other type of communication strategy observed is using mesh topology among the swarm of robots. In this type of communication topology, every device will share the ultrasonic sensor data with every device[20]. In this paper, three NodeMCU ESP8266 are used to share data of ultrasonic sensors among themselves. Here, each device will perform the following operations: a) its operation, b) monitoring the operation of other devices, and c) sending its data to other devices. A block diagram of this type of communication among the swarm is given in Fig. 7.

The ESP-NOW protocol is used to communicate among the different NodeMCU using the MAC address of the devices. One of the main advantages of using this method is that it does not require network availability, i.e., it can perform an assigned operation independently of its location. Also, using this swarm method, it can easily monitor the operation of swarm[2,4]. Apart from this, there are some disadvantages of using this swarm technique, such as that it can only operate on a small scale and in a small area, and there might be a collision when any bot replaces the other bot in case of malfunctioning.

5 Communication among Swarm Using Ring Topology

The third type of communication strategy studied in this paper is ring topology. In this type of communication topology, one device can share its ultrasonic sensor data with one of the adjacent devices and so on, then will finally attach to the first device, which eventually forms the ring structure. Here, three NodeMCU ESP8266 are used to share data of ultrasonic sensor among themselves, and each device will perform the operation: a) its operation, b) monitor the data device whose data is coming, and c) send its the data to an adjacent device. A block diagram of this type of communication among the swarm is given in Fig. 8.

In our study, we explored the use of low-cost NodeMCU ESP8266 devices to establish master-slave communication in swarm robotics through the Internet of Things (IoT), evaluating three network topologies: star, mesh, and ring. We set up a simple swarm system consisting of three robots, with each topology implemented under consistent conditions. The star topology featured one master node communicating directly with each slave node, providing centralized control. The mesh topology involved all nodes being interconnected, allowing for peer-to-peer communication, while the ring topology connected nodes in a sequential manner, where each node communicated only with its immediate neighbors. We measured performance based on latency, throughput, and error rate. Results showed that the star topology was the most efficient, with the lowest average latency at 15ms, compared to 30ms for the ring and 45ms for the mesh topology.

It also achieved the highest throughput and the lowest error rate, indicating superior communication efficiency and reliability. The star topology's centralized nature reduced communication delays and packet losses, making it highly suitable for time-

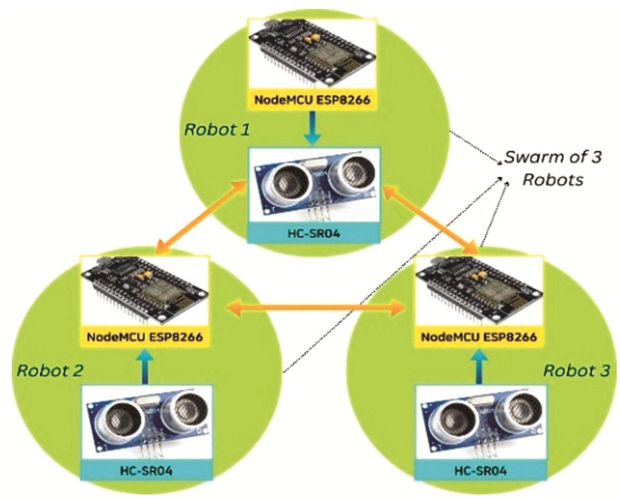

Fig. 7 — Block Diagram of Mesh Topology among Swarm

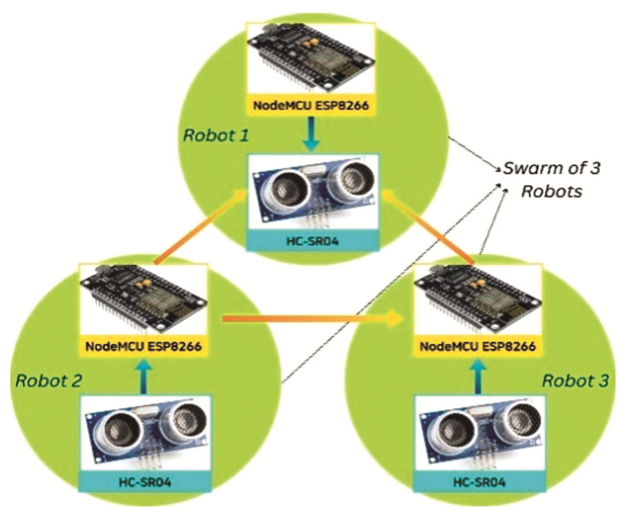

Fig. 8 — Block Diagram of Ring Topology among Swarm

sensitive tasks. In contrast, though robust and fault-tolerant, the mesh topology introduced higher communication overhead and delays, while the ring topology exhibited longer delays due to its sequential data transfer process. Our findings underscore the practical advantages of the star topology and the NodeMCU ESP8266 devices, highlighting their potential for cost-effective solutions in swarm robotics. This approach offers significant cost benefits without compromising performance, making it a viable option for agriculture, manufacturing, and construction applications.

6 Results and Discussion

Our study investigated low-cost NodeMCU ESP8266 devices to establish master-slave communication in swarm robotics using IoT, utilizing different network topologies: star, mesh, and ring. Our experiments with a simple swarm system of three robots revealed that the star topology was the most efficient for master-slave communication. This topology allowed for smooth and reliable data transfer, enabling the master node to effectively monitor and control the slave nodes, facilitating tasks such as aggregation, pattern formation, and coordinated motion. Numerical analysis showed that the star topology had the lowest average communication time at 15ms, compared to 30ms for the ring and 45ms for the mesh topology. As shown in Fig. 9, the star topology exhibits the lowest latency, followed by the ring and mesh topologies.

The star topology outperformed both the mesh and ring topologies regarding communication efficiency and reliability. The centralized nature of the star topology reduced communication delays and data packet losses, making it highly suitable for time-sensitive tasks. In contrast, while offering robustness and fault tolerance, the mesh topology introduced higher communication overhead and potential delays due to its decentralized nature.

In Fig. 10 it can be observed that the star topology achieves the highest throughput, with ring and mesh topologies showing slightly lower values. The ring topology, which provided a balanced communication load, exhibited longer delays due to the sequential data transfer process. These findings highlight the practical advantages of the star topology over traditional mesh networks, particularly in applications requiring rapid and efficient communication.

Our comparative analysis with existing methods further underscores the advantages of using

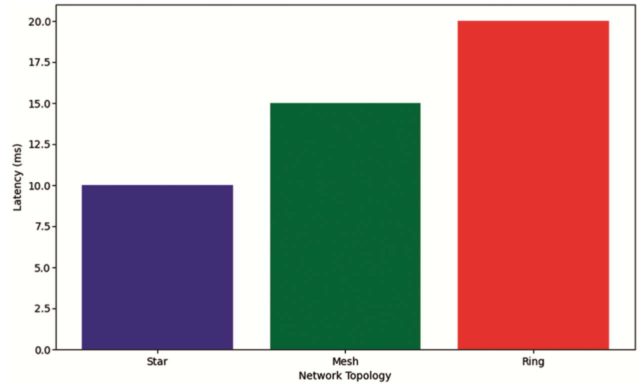

Fig. 9 — Latency of different network topologies

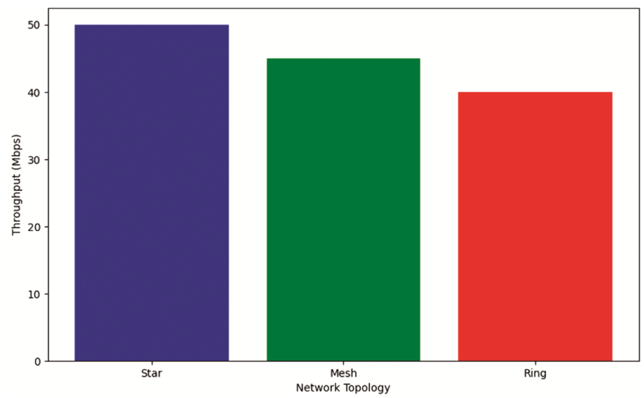

Fig. 10 — Throughput of different network topologies

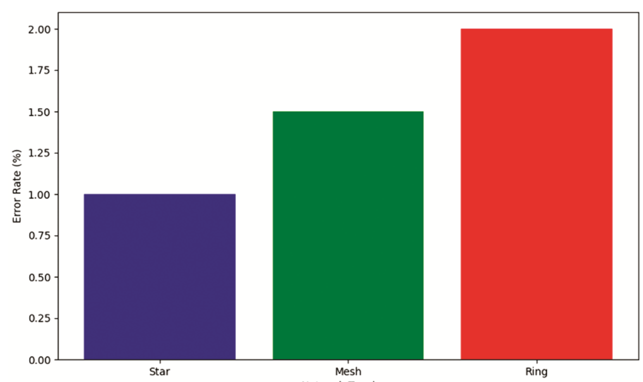

Fig. 11 — Error rate of different network topologies

NodeMCU ESP8266 devices. Traditional swarm robotics often rely on more expensive and less accessible technologies, limiting their applicability in cost-sensitive scenarios. Fig. 11 shows the star topology has the lowest error rate, while the ring topology has the highest. Using NodeMCU ESP8266 devices offers a significant cost advantage without compromising performance, making swarm robotics more accessible and feasible for a broader range of applications. This approach has important implications for agriculture, manufacturing, and construction industries, where affordable and efficient communication systems can enhance productivity and reduce costs. By employing low-cost devices, our study demonstrates the potential for innovative and cost-effective solutions in swarm robotics, paving the way for future research and development in this field.

7 Conclusion

Our study demonstrates the feasibility of using low-cost NodeMCU ESP8266 devices to establish efficient master-slave communication in swarm robotics through IoT. By evaluating various network topologies, our results indicate that the star topology provides the most effective communication and coordination among robots. Our approach offers a viable alternative to traditional communication methods and paves the way for broader adoption of swarm robotics in various applications.

Future research could explore several avenues to build upon our findings. For instance, investigating the integration of other low-cost IoT devices and sensors could further enhance system capabilities and versatility. Additionally, examining the performance of the star topology in more extensive and complex swarm configurations could provide deeper insights into its scalability and robustness. Comparative studies with emerging communication technologies and optimization algorithms may offer valuable perspectives on improving swarm robotics performance and efficiency. Our study highlights the potential of leveraging low-cost technologies for advancing swarm robotics, contributing to a more accessible and innovative field with broad applications across diverse industries.

References

1 Clerigues D, Wubben J, Calafate C T, Cano JC & Manzoni P, *EURASIP J Wirel Commun Network*, 39 (2024) ISSN 1687-1499.
2 Huang S H, Huang Y H, Blazquez C A & Chen C Y, *Adv Eng Inform*, 51 (2022) 101536.
3 Agrawal A K, Pidge P, Bharti M, Dev M P & Kedare P K, Innovations and future of robotics. *In Advances in Manufacturing and Industrial Engineering*: Select Proceedings of ICAPIE 2019, (2021) 975.
4 Navarro I & Matía F, An introduction to swarm robotics, ISRN Robotics, (2013) 1 ISSN 2090-4860.
5 Bento A C, *Int J Adv Res Comput Sci Manag Stud*, 6 (2018).
6 Liekna A & Grundspenkis J, Towards practical application of swarm robotics: Overview of swarm tasks. Engineering Rural Development, 13 (2014) 271.
7 Bharti M & Agrawal A K, War field spy and fighter robot. In Recent Advances in Mechanical Engineering: Select Proceedings of NCAME 2019. (2020) 565.
8 Sivakumar T B, Hussain S H, Kanmani A & Babu M H A, Surveillance robot for health care applications using IoT and wireless sensor network, (2021).
9 Sharkey A J C & Sharkey N, *Adv Appl Art Intell*, (2006) 157.
10 Sangeetha B P, Kumar N, Ambalgi A P, Haleem S L A, Thilagam K & Vijayakumar P, *Sustain Energy Technol Assess*, 52, (2022) 101973.
11 Agrawal A K & Bharti M, Aspects and use of digital agriculture using IoT. In Harnessing the Internet of Things (IoT) for a Hyper-Connected Smart World, *Apple Academic Press*, (2022) 243.
12 Albani D, Jsselmuide I J, Haken R & Trianni V, Monitoring and mapping with robot swarms for agricultural applications. In 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), (2017) 1.
13 Gazi P, Jamshidi M, Jevtic A & Andina D, A mechatronic system design case study: Control of a robotic swarm using networked control algorithms. In 2010 IEEE International Systems Conference, (2010) 169.
14 Agrawal A & Padhi P K, Low cost approach to a two-agent imitating master-slave robotic system, In 2021 International Symposium of Asian Control Association on Intelligent Robotics and Industrial Automation (IRIA), (2021) 414.
15 Agrawal A K & Bharti M, Feature, technology, application, and challenges of Internet of Things. In Big Data, IoT, and Machine Learning, *CRC Press*, (2020) 255.
16 Bhuiya A, Mukherjee A & Barai R K, Development of Wi-Fi communication module for ATmega microcontroller based mobile robot for cooperative autonomous navigation, In 2017 IEEE Calcutta Conference (CALCON), (2017) 168.
17 Wang X, Mao X & Khodaei H, *J Build Perform*, 33, (2021) 101603.
18 Ahmed E Q, AlJazaery I A, Al-Zubidi A F & Alrikabi H T S, *Period Eng Natural Sci*, 9 (2021) 409.
19 Jasim N A & Al Rikabi H T S, *Int J Interact Mobile Technol*, 15 (2021).
20 Shlibek M & Mhereeg M, Comparison between Arduino based wireless and wire methods for providing power theft detection, *Eur J Eng Sci Technol*, 2 (2019).
21 Yaichi M, Rebhi M & Bousmaha B, Intelligent unidirectional road lighting control using NodeMCU ESP8266. In Artificial Intelligence and Renewables Towards an Energy Transition, (2021) 682.
22 Thayer T C, Vougioukas S, Goldberg K & Carpin S, *IEEE Trans Automat Sci Eng*, 17 (2020) 1184.