2022 13th International Conference on Information, Intelligence, Systems & Applications (IISA) | 978-1-6654-6390-4/22/$31.00 ©2022 IEEE | DOI: 10.1109/IISA56318.2022.9904339

WiCHORD: A Chord Protocol Application on P2P LoRa Wireless Sensor Networks

Christos-Panagiotis Balatsouras, Aristeidis Karras, Christos Karras, Dimitrios Tsolis, Spyros Sioutas
Decentralized Systems Computing (DSC) Group
Computer Engineering and Informatics Department
University of Patras, 26504 Patras, Greece
{balatsouras, akarras, c.karras, sioutas}@ceid.upatras.gr, dtsolis@upatras.gr

***Abstract*—On the modern era of Internet of Things (IoT) and Industry 4.0 there is a growing need for reliable wireless long range communications. LoRa is an emerging technology for effective long range communications which can be directly applied to IoT applications. Wireless sensor networks (WSNs) are by far an efficient infrastructure where sensors act as nodes and exchange information among themselves. Distributed applications such as Peer-to-Peer (P2P) networks are inextricably linked with Distributed Hash Tables (DHTs) whereabouts DHTs offer effective and speedy data indexing. A Distributed Hash Table structure known as the Chord algorithm enables the lookup operation of nodes which is a major algorithmic function of P2P networks. In the context of this paper, the inner workings of Chord protocol are highlighted along with an introduced modified version of it for WSNs. Additionally, we adapt the proposed method on LoRa networks where sensors function as nodes. The outcomes of the proposed method are encouraging as per complexity and usability and future directions of this work include the deployment of the proposed method in a large scale environment, security enhancements and distributed join, leave and lookup operations.**

***Index Terms*—Internet of Things, Wireless Sensor Network, LPWAN, LoRa, Chord, Distributed Hash Tables, P2P**

I. Introduction

In the modern everyday life and also in the modern industry, there is more and more need for interconnected devices which are useful to support the human activity. From now on with the Internet of Things (IoT) technology, not only personal computers but also any other device may be capable of accessing the internet. IoT technology can provide the user with a variety of applications such as environmental conditions monitoring of far away places via the use of a wireless sensor network, or controlling and operating smart connected devices remotely.

The purpose of IoT is to bring the internet connectivity from personal computers to any other device, in order to achieve more process automation and lower human intervention especially on simple processes [1], [2]. An Internet of Things application consists of a network of devices that access the Internet and exchange information among themselves. This network may be a wireless sensor network, a network of smart household devices on a smart home application, a network of equipment connected to the internet on a smart industry environment or a network of sensors and connected cars on a smart city application and more [3]. These devices that support an IoT network are low-power embedded devices such as Microcontroller Units (MCUs) and also network enabled smart devices such as wearable devices, surveillance cameras, smart sensors, smart appliances and more. Some of the most notable purposes of these IoT devices and networks are data acquisition from sensors and remote device control over the internet.

978-1-6654-6390-4/22/$31.00 ©2022 IEEE

The IoT application that will be discussed on this work is the usage of IoT technology on Wireless Sensor Networks (WSNs). The main purpose of Wireless Sensor Networks is the monitoring of conditions from the local environment that each node is being placed and with the IoT technology, the data from the sensors can be available to users remotely. These networks consist of nodes randomly distributed in space or placed in positions based on a predefined topology. Each node is a device mostly equipped with a Microcontroller Unit, various sensors, a transceiver module and a power source [4]. The communication between nodes on this kind of networks is either between two sensor nodes or from one sensor node towards the base station.

Various wireless networking technologies are used in order to connect the sensor nodes on a Wireless Sensor Network. Depending on the environment that the sensor network is deployed or the energy consumption of the network, the nodes are using short range wireless communication technologies such as Wi-Fi, Bluetooth and Zigbee or long range wireless connectivity technologies such as LoRa and other LPWAN technologies, in order to communicate with each other. On this work, the LoRa wireless communication technology will be selected (among others) for the communication between the nodes on a sensor network. Some key features of LoRa technology that are making it more suitable for wireless sensor network communication are the following. First, LoRa supports long range communication in which two nodes can be kilometers apart, thus supporting wider area sensor networks [5]. Also, this technology has very low energy consumption, resulting to long lasting battery life per node which is very important to sensor networks deployed on environments without a continuous electricity supply [5]. A comparison of LoRa technology between other long range wireless technologies is being done on the Table I. We select LoRa as the optimal network technology for the Chord implementation due to network topology, signal resilience and simultaneously license-free deployment, which is important given the high cost of licence of similar technologies. For this case, LoRa is the superior technology in terms of cost-effectiveness, deployment, and network topology.

On this work, a new method for data packet routing and communication between the nodes on a wireless sensor network is proposed. This method is based on the Chord Peer-to-Peer (P2P) Protocol [13]. Originally, Chord is a Distributed Hash Table in which, using consistent hashing, each key is mapped to a specific hash value on a fixed m-bit circular hash space [13]. The basic idea of Chord Protocol is to lookup for a key and return the responsible node for that key in particular, on a dynamic network in which nodes are joining or leaving the network while the network remains functional [13]. On a P2P network using the Chord protocol, the computational cost for a query to be resolved is a logarithmic function of the total number of nodes on the network [13]. The main characteristics of Chord Protocol that are making it suitable for a Wireless Sensor Network use case are the following. First, the continuous network operation with nodes joining or leaving the network is very useful on sensor networks, because various sensor nodes may join or leave the network dynamically. For example, a sensor node will leave the network if it runs out of battery or will join the network again if battery is recharged. Also, the P2P network concept is suitable on a wireless sensor network in which two nodes might be physically out of range, so they will establish a connection between themselves using other intermediate nodes of the network.

The primary research objective of this paper is the incorporation of the concept of the Chord Protocol from Distributed Hash Tables on P2P Wireless Sensor Networks using LoRa communication technology for Internet of Things (IoT) applications.

The remaining of this work is structured as follows. On section II, a review on the literature about related work in the fields of P2P systems, Distributed Hash Tables, Wireless Sensor Networks and regarding the adaptation of the Chord Protocol on Wireless Sensor Networks is done. On section III, the proposed method of applying the Chord Protocol on a Wireless Sensor Network with LoRa technology is described. On section IV, the results from the simulation of the suggested method on a virtual sensor network are presented. Finally, on section V the conclusions of this work are presented.

II. Previous Work

Peer-to-peer (P2P) systems have long been introduced in the scientific area [14], [15]. Applications of P2P systems include routing [16], load balancing [17] and security optimizations [18]. P2P systems are also utilized for managing the supply chain in the agricultural industry as outlined in [19].

Previous works in P2P systems including distributed frameworks [20], [21] or dynamic query processing [22] are the foundations for research directions in P2P systems. While in [23], [24] decentralized digital content exchange and copyright protection via P2P networks are presented. Also, efficient energy-level queries are presented in [25], while schemes for web service discoveries are introduced in [26], [27].

Distributed hash tables are also inextricably linked with P2P systems, whereabouts DHTs are designed for fast data indexing in big wired networks [28]. A DHT is a distributed system that offers a lookup service similar to a hash table, where key-value pairs are stored within a DHT and any participating node can obtain the value associated with a specific key quickly and effectively [29].

Innovative Applications of the Chord protocol for Wireless Sensor Networks also include localization schemes [30], [31], hierarchical methods for key storage and lookup [32], as well as efficient query processing in large scale systems [33].

Additionally, in [34], an intriguing method for implementing the Chord Protocol on a Wireless Sensor Network is proposed. In this method, a randomly distributed sensor network is structured into sensor node clusters, depending on the physical position of each sensor node, with the most powerful node in terms of energy levels being chosen as the cluster leader. The head sensor nodes of each cluster on the sensor network operate as nodes on the Chord Protocol hash space and data forwarding on the sensor network begins with the simple nodes from each cluster and proceeds to the head of each cluster and then to the base station. Additionally, another method by the same authors as [34] can be found in [35] where an energy-based chord implementation is presented.

To connect wireless sensor networks to existing networks over the Internet, [36] suggests using a Chord Protocol overlay. The Chord Protocol lookup query is the main function of the proposed solution. The sensor network's nodes are divided into master and slave nodes, after being given a unique identification by hashing their physical addresses. This means that each master node keeps a finger table of local master and slave nodes to forward queries to the appropriate node. Using metadata as keys, each node's hash space is indexed.

In [37], a chord Protocol-based routing on wireless sensor networks is shown. The chord protocol nodes' topology is virtual and based on relative distances between surrounding nodes, with each node's successor and predecessor close by. There are short paths between local nodes on the actual architecture rather than the virtual topology, which reduces communication costs. On joining the proposed Chord network, a node sends a "hello" message, and it's neighbours update their node lists. The lookup operation of this protocol sends the query to the responsible node from its neighbours. A network node that ceases broadcasting messages is considered to have failed and disconnected.

Another approach presented in [38], relates to the construction of a new chord protocol overlay for wireless sensor networks with an improved stabilisation function and enhanced data packet exchange on the network. The suggested approach is connected to the construction of a new chord protocol overlay on wireless sensor networks, with optimised stabilisation function and enhanced interchange of data packets on the network. Finally, alternative versions of Chord protocol as in [39], are capable of self-adaptation in P2P overlay networks.

TABLE I

COMPARISON OF LPWAN TECHNOLOGIES (COMPILED FROM [6] [7] [8] [9] [10] [11] [12]).

Parameter	LoRa	SigFox	NB-IoT	LTE-M	DASH7
Standard	LoRa Alliance	SigFox / ETSI LTN	3 GPP Release 13,14	3 GPP	Dash Alliance
Bandwidth	250 kHz	100 Hz	200 kHz	1.4-20 MHz	433/868/915 MHz
Modulation	FSS/CSS	D-BPSK	QPSK	DL: OFDMA, 16 QAM	GFSK
Spectrum	1175 kHz	200 kHz	200 kHz	Licensed LTE bands	Licensed
Frequency band	EU : 868MHz	EU : 868MHz	$7-900$MHz	Cellular Band	Cellular Band
Transmission	FHSS (Aloha)	UNB	FDD	FDD/TDD	BLAST
Topology	Star-of-stars	Star	Star	Star	Half
Security	AES 128b	Optional encryption	NSA AES 256	AES 256	AES 128
Range (Urban)	$2-5$ km	$3-10$ km	$1-5$ km	$1-5$ km	1 km
Range (Rural)	20 km	50 km	$10-15$ km	$10-15$ km	2 km
Data Rate (Min)	250 bps	100 bps	100 $kbps$	1 $Mbps$	27.8 $kbps$
Data Rate (Max)	50 $kbps$	600 bps	200 $kbps$	4 $Mbps$	200 $Kbps$
Throughput	50 $kbps$	-	127 $Kbit$	1 $Mbit$	167 $kbit$
Energy Consumption	Very Low	Low	Medium Low	Medium	Low
Battery Life	$\sim$ 10 years	$\sim$ 12 years	$\sim$ 10 years	$\sim$ 2 years	$\sim$ 10 years
Deployment Cost	Moderate	Moderate	High	High	Moderate

III. PROPOSED METHODOLOGY

In this section of this paper, the proposed solution of utilizing the Chord Protocol on a P2P Wireless Sensor Network using LoRa communication technology is developed. Initially, the architecture of the proposed Wireless Sensor Network model that will be used on this application is described. Subsequently, the modified Chord Protocol that is proposed for the use case of this work will be presented and the operations of this new custom version of Chord will be described in an algorithmic manner. Finally, the adaptation of the new modified Chord Protocol on LoRa P2P Wireless Sensor Networks will be discussed.

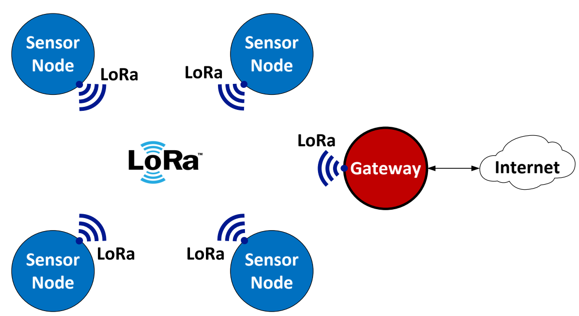


Fig. 1. The architecture of the proposed WSN model

A. Selected Wireless Sensor Network Model

Definition 1 (Sensor Node)**.** A Sensor Node is a simple node on the Wireless Sensor Network, consisting of a Microcontroller (MCU) board equipped with some sensors capable of measuring data from the local environment and forwarding them to the base station.

Definition 2 (Gateway or base station)**.** A Gateway or base station is the node of the Wireless Sensor Network which is connected on the Internet and collects all the data from all the Sensor Nodes of the network.

The architecture of the proposed Wireless Sensor Network model to support the application that is presented on this work is the following. Initially, we consider a network which consists of Sensor Nodes, as stated on the definition 1, that are deployed randomly in space on a field and each of them is capable of measuring useful parameters using sensors attached to a micro-controller, in order to transmit them as data packets to a gateway connected to the Internet.

The Sensor Nodes on this network are utilizing the LoRa communication technology to establish a link among other nodes on the network and exchange information. Each Sensor Node on this network is only aware of a few other nodes of the network and has no prior knowledge of any further nodes from the network in which it belongs to. On the other hand, all the Sensor Nodes of the network on the physical layer are within the LoRa range of the Gateway by design, and they are capable of communicating with each other at any possible combination.

At any given time, only one of the Sensor Nodes on the network is elected to act as the base station or gateway, which as described on the definition 2, gathers all the sensor data from all the other nodes on the network and broadcasts them on the Internet for further data processing. However, in order for the network to be robust in a case of the gateway node failure, all the other Sensor Nodes have the ability to be elected as base stations. Hence, if the gateway node fails, one of the Sensor Nodes of the network will be elected as base station to preserve the Wireless Sensor Network operation. A schematic of the proposed sensor network model architecture is show in the Figure 1.

B. Modified Chord Protocol for Wireless Sensor Networks

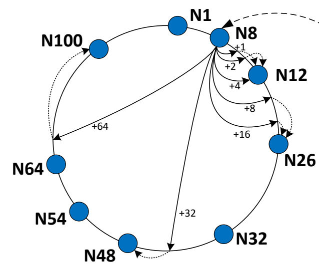
In this subsection, the modified Chord Protocol for P2P Wireless Sensor Networks is described theoretically, highlighting the interventions done on the classic Chord Algorithm from [13] and also in comparison with the original Chord Protocol from Distributed Hash Tables as shown in [13].

The modified Chord Protocol proposed in this work has the following usability, in order to solve the following problems.

First and foremost, in a Wireless Sensor Network environment the nodes may connect to the network or disconnect from it at any moment because their power source relies on batteries or for any other reason. So, the network is dynamic regarding the set of active nodes each time and because of this the Chord Protocol, with the Join and Leave operations that it supports, can contribute in maintaining the network's uninterrupted operation and stability while at the same time several Sensor Nodes can join or leave the network. Next up, the main motivation to adapt the Chord Protocol on Wireless Sensor Networks is to give the ability of supporting lookup queries about the search for a node in the network. Specifically, these lookup queries are able to answer the question of whether a node with a specific unique identification value exists on the network or not, by using the Lookup operation of the Chord Protocol to search for a node by given identification value.

The structure of this modified Chord Protocol overlay for Wireless Sensor Networks is as follows. First, same as the initial Chord Protocol proposed in [13], we consider a fixed circular m-bit hash space containing 2^m integer addresses in the range $[0, 2^m)$ with values increasing clockwise on the circle. Each Sensor Node on the sensor network has the functionality of a node in the Chord network and is receiving a unique identification number (ID) from the hash space by hashing the unique MAC address of the node using the "SHA-1" algorithm. The ID of a node n is calculated as SHA-1(n.MAC_Address) mod 2^m, where the number 2^m is the number of maximum available addresses in the circular hash space. In contrast to the original Chord, the ring shaped hash space can map only the IDs of the nodes of the network and there are no keys to be assigned to each node such as the original Chord suggests. However, the lookup operation in this Chord approach returns the successor node to a given value the same as the initial Chord Protocol, but in this Chord application there are only nodes to be looked up in the network and no keys. So, each Lookup query of this application is only concerned of locating a node within the network by finding the responsible node for the ID resulted from the hashing of the node's MAC address.

Every node on the network, like the original Chord Protocol from [13], is only aware of it's immediate successor and predecessor on the Chord ring and is also linked with a small subset of the nodes belonging to the network, in which the nodes are also known as "fingers". This subset of nodes that are in direct connection to the node in the Chord ring is stored in the node on a table of m entries, known as the "Finger Table". The calculation of i-th entry of the Finger Table for a node n is performed by looking up the corresponding node for the value of $n+2^i$, with the integer i in the range $[0, m)$. The first item of the finger table of node n is the one with $i = 0$ and it is the n's immediate successor, while the i-th item of the table is a node of the Chord ring with a distance of at least 2^i positions ahead from the node n. In conclusion, a node keeps ties to its successors from the finger table, with the first finger being the node's direct successor, and is also linked with the it's immediate predecessor on the circle. Keeping a successor list is similar to that of the Pastry leaf set (to route in "slow" mode when there are no other choices) [40].



Finger Table of Node N8	
N8 + 1	N12
N8 + 2	N12
N8 + 4	N12
N8 + 8	N26
N8 + 16	N26
N8 + 32	N48
N8 + 64	N100

Fig. 2. Chord ring network consisting of Sensor Nodes alongside with the Finger Table of one of the nodes, inspired from [13]

An example of the circular structure of the hash space of the Chord Protocol with $m = 7$ bits, is shown on the figure 2. In this example some nodes are already joined in the network and the finger table is displayed for one of these nodes.

The functionality of this Modified Chord Protocol for Wireless Sensor Networks relies on the following operations. First of all, regarding the structure of the network in which a node may connect to the network or disconnect from it, we consider that a node is capable of executing the "Join" and "Leave" operations on the network respectively. Next up, in order for a node to resolve a lookup query, the operations of "find_successor" and "find_closest_predecessor" are considered as auxiliary operations to support the lookup query operation functionality. All the above mentioned operations and methods will be discussed in the next paragraphs.

The auxiliary methods "find successor" and "find closest predecessor" as introduced in [13], are supporting the main Chord protocol functionality, which is to find the next possible corresponding node to a given value. Like the original Chord, these functions have the same operation in the modified Chord protocol application. In order for the aforementioned auxiliary operations and the modified Chord as well to function properly and return correct results, the properties of the Chord ring hash space structure must be preserved for the satisfaction of the principle that every search key is mapped to it's successor node.

When a new node is requesting to join an already existing Chord ring network, the next steps are followed. First, the identification number (ID) of this node in the network is calculated by hashing the unique MAC address of this node device using the "SHA-1" algorithm as mentioned above. Similar to the original Chord in [13], in order to introduce a new node into a network, the finger table of this new node is populated with the corresponding nodes as the Chord structure suggests by asking an other node of the network to lookup for the responsible nodes of certain values as mentioned in a previous paragraph. Also, as the original Chord suggests, when a new node n joins the network, a set of nodes on the regions $[n - 2^i, n.predecessor - 2^i + 1]$, with $i \in [0, m)$, of the ring network must be updated in order for their finger tables to include the new joined node n. Unlike the original Chord, in this modified version of it, when a new node joins a network there are no keys to be transferred to the new node. Regarding the complexity of the operation when node n joins in a N-node Chord network, same as the initial Chord in [13], $O(\log N)$ nodes have to update their finger tables. The join operation is presented on the algorithm 1.

Algorithm 1 Join operation of a node on the Modified Chord

Require: An established Chord Wireless Sensor Network with m-bit fixed hash space and nodes belonging in the N set, a new node n requesting to join the network
Ensure: The same network with the node n joined in and all the required nodes updated about the existence of n

Set n.ID $\leftarrow$ SHA-1(n.MAC_Address) mod 2^m
Set $N \leftarrow N \bigcup \{n\}$, Set $|N| \leftarrow |N| + 1$
for $i = 0$ to $i < m$ **do**
    n.finger_table[i] $\leftarrow$ n'.find_successor($n + 2^i$) {A random node $n' \in N$ is asked to find the successor node to each value}
end for
Set n.successor $\leftarrow$ n.finger_table[0]
Set $s \leftarrow n$.successor {s is the successor of n}
Set n.predecessor $\leftarrow$ s.predecessor {n resides between s and it's predecessor, hence the s.predecessor is now the predecessor of n}
Set s.predecessor $\leftarrow n$ {Notify s that it's new predecessor is n}
Set $p \leftarrow n$.predecessor {p is predecessor of n}
Set p.successor $\leftarrow n$ {Notify p that it's new successor is n}
for $i = 0$ to $i < m$ **do**
    for each node $n' \in [n - 2^i, p - 2^i + 1]$ **do**
        Ask from n' to update it's finger table
    end for
end for

In this modified Chord Protocol approach for Wireless Sensor Networks, when a Sensor Node n is intended to leave the network either because it's battery is about to run out or the user wants to withdraw this node from the network the following process is taking place. First, the node to leave notifies it's successor and predecessor nodes about it's forthcoming departure from the network, in order to update their predecessor and successor node pointers respectively. Also, the set of nearby preceding nodes that was notified to update their finger tables during the join of node n on the network, is asked to update their finger tables in order to remove the departing node n from their entries. Unlike the original Chord from [13] where each node maintains a list of backup nodes if a query failure happens due to an unavailable node and periodically all nodes refreshing their successor and predecessor pointers, in this modified approach the node's "Leave" operation is considered a separate function. Regarding the computational cost whenever a node n disconnects from the N-node Chord network, because of the implementation of the new "Leave" method which notifies all the affected nodes to update their finger tables the same way as the "Join" operation does, $O(\log N)$ nodes are updated. The functionality of the distinct node leave operation is described on the algorithm 2.

Algorithm 2 Leave operation of a node on the Modified Chord

Require: An established Chord Wireless Sensor Network with m-bit fixed hash space and nodes belonging in the N set, a node $n \in N$ intended to leave the network
Ensure: The same network without the node n and all the required nodes updated about the withdrawal of n

Set $N \leftarrow N - \{n\}$, Set $|N| \leftarrow |N| - 1$
Set $s \leftarrow n$.successor {s is the successor of n}
Set $p \leftarrow n$.predecessor {p is the predecessor of n}
{Update successor, predecessor of n's direct neighbors}
Set p.successor $\leftarrow s$
Set s.predecessor $\leftarrow p$
for $i = 0$ to $i < m$ **do**
    for each node $n' \in [n - 2^i, p - 2^i + 1]$ **do**
        Ask from n' to update it's finger table
    end for
end for
Set n.finger_table $\leftarrow$ Null, Set n.successor $\leftarrow$ Null, Set n.predecessor $\leftarrow$ Null

The lookup query in the modified Chord of this paper, in difference with the lookup operation proposed in [13], serves only the purpose of finding a node of the network by it's MAC Address. In order to find the corresponding node to an input key on a lookup query, same way as the initial Chord, the input key is hashed by the "SHA-1" algorithm and the lookup value derives from the calculation of SHA-1(key) mod 2^m, with 2^m being the maximum available addresses in the hash space of the network. Afterwards, the lookup value is passed as input to the "find_successor" function to determine the responsible node for this query. The modified lookup operation is explained on the algorithm 3.

Regarding the complexity of the lookup query operation, the lookup message will ultimately reach the target node since each step advances the key. Each node in a N-node Chord ring network stores a m-entry finger table, hence the per node's finger table size is fixed at $O(m)$, in the sense that m does not vary due to the use of consistent hashing. Experiments indicate that the predicted length of a routing path is $\frac{1}{2} \log N$. The length of a routing path is most likely to be $O(\log N)$. This may be intuitively explained by noting that a lookup query for a distant node travels relatively quickly at first, nearly half the distance with each step. Since m is constant, we may assume that m is selected to be large enough to "cover" any possibly large N, resulting in a significant amount of redundant storage when N is small.

C. Adapted Chord Protocol in WSNs powered by LoRa

In this part of the proposed methodology of this work, the modified Chord Protocol for P2P Wireless Sensor Networks which presented in the previous subsection, is adapted to

Algorithm 3 Lookup Query operation of the Modified Chord

Require: An established Chord Wireless Sensor Network with m-bit fixed hash space and nodes belonging in the N set, an input key being a MAC address of a potential node to lookup

Ensure: The node n with MAC address same as the requested or Null in case of this node does not exists.

```
lookup_value ← SHA-1(key) mod 2^m
if n < lookup_value then
    result_node ← n.find_successor(key)
else
    Consider n_0 as the first node of the network
    result_node ← n_0.find_successor(key)
end if
```

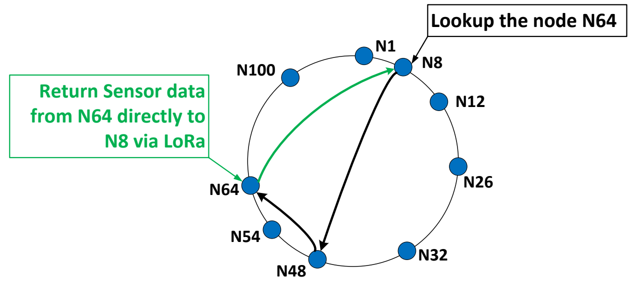


Fig. 3. a lookup message from one node to an other on a Chord ring network consisting of Sensor Nodes

Wireless Sensor Networks powered by the LoRa wireless networking technology.

Regarding the implementation of the nodes in hardware for this LoRa WSN, each Sensor Node of the Wireless Sensor Network consists of a board with a Microcontroller Unit (MCU) such as the ESP32 from Espressif Systems and a chip to support LoRa communication alongside with the suitable antenna. The node is equipped with sensors connected to the board for data acquisition and monitoring purposes. These sensors are capable of measuring a variety of metrics regarding the local environment such as temperature, humidity, rainfall, the existence of fire or any other environmental condition depending on the application on which the sensor network is used. After the data collection of the sensors, the node transforms the data to packets and broadcasts them towards the gateway node of the network. The node working as a gateway at any given time has the same hardware implementation as the regular Sensor Node and also has enabled Internet connection, in order to broadcast the data received from the other nodes to any supported Cloud Application for further data processing.

As mentioned on a previous subsection regarding the architecture of the WSN model used in this work, a network of randomly deployed Sensor Nodes on a field is considered, in which all the Sensor Nodes are within the LoRa range in order for every node to be able to establish a direct link with any other node of the network. However, a node keeps ties only to some other nodes in order to resolve a node lookup query as the Chord Protocol dictates and as mentioned in the proposed method from the previous subsection. So, each node is only aware of it's successor and predecessor nodes alongside with the nodes in it's finger table. Also, some fundamental specifications of the Chord network such as the ID of the ring network and the size of the m-bit hash space of the Chord structure are known to every node on the network.

In the Modified Chord for LoRa Wireless Sensor Networks, after a lookup query or any other message reaches the destination node, the reply message of the node with it's sensors measurements data is included in a package and this is forwarded directly to the node that performed the query in $O(1)$ time, because the nodes are within the LoRa network range and both the identity of the sender and the receiver are now known after the completion of the lookup query. The figure 3 presents an example of a lookup query performed by one node acting as the gateway on a Chord Sensor Network, in order to lookup if a certain node exists on the network and to get the reply with the sensor data from that node.

On this modified Chord network implemented with LoRa technology, when a node is ready to join the network, it broadcasts a "hello" message with it's MAC address to an other random node of the network or to the gateway, in order for the "Join" operation to be performed and the node to join the Chord sensor network.

When the battery of a Sensor Node is about to run out or the network administrator wants to withdraw a Sensor Node from the network, the node sends a "departure" message to any other node of the network or towards the gateway, in order for the "Leave" subroutine to be executed on the network, to remove the node and update the successor, predecessor pointers and finger tables of all the other affected nodes on the network.

In order to build from scratch a Chord WSN using LoRa technology with the proposed modified Chord model of this work, a new "Build" operation is considered. This operation is executed from the gateway node during initialization and scans the environment for "hello" messages transmitted by the nodes. Then the "Join" operation is activated to complete the integration of the nodes in the network, while is asked from all the nodes in the network to update their finger tables as the join function suggests. The functionality of the network build operation is presented on the algorithm 4.

IV. Results

In order to test the functionality of the new Modified Chord Protocol for WSNs that was proposed above and to verify it's complexity with experiments, a simulator application was developed using the Python 3.9 programming language. The development environment that was preferred is the "PyCharm 2022.1.1". Also, in order to create the simulator, the python packages of "hashlib", "random", "secrets" and "csv" were imported. The "hashlib" package was selected in order to implement the "SHA-1" algorithm for the purpose of hashing a value whenever is needed in the Chord Protocol. The "secrets"

Algorithm 4 Operation to Build the Chord LoRa WSN

Require: A Sensor Node b functioning as the gateway of an empty network N, several Sensor Nodes $s \in S$ sending "hello" messages

Ensure: A built Chord Wireless Sensor Network with nodes belonging in the N set

```
frequency ← 868 MHz
Scan for MAC addresses of each node s ∈ S broadcasting "hello" message in 868 MHz and requesting to join the network N.
for all Nodes s ∈ S requesting to join the network N do
    s.join(N)
end for
```

python module is useful to generate random MAC addresses for the simulation of the Wireless Sensor Network and the "random" library is used to select a random node of the network, to be used as the requested node for the simulation of the lookup query functionality.

In the created simulator application of the Modified Chord, the Sensor Nodes of the network are represented by a Python class, containing all the attributes of the Sensor Node as an entity. Also, the operations of Chord that every Sensor Node executes, such as "Join", "Leave" and "Lookup" are implemented as methods of that Python class. Furthermore, the Sensor Network in which the Sensor Nodes belong, is also represented by a Python class which contains all the network's fundamental details such as the size of the m-bit hash space. When the simulator application of the Modified Chord starts running, a set of Sensor Node objects with random generated MAC Addresses is compiled. Afterwards, the network build operation that was mentioned in the previous section is called to build the Chord Sensor Network. After the completion of the network build operation, the simulator is ready to accept user queries to lookup for nodes of the network and is capable of measuring the number of hops between the nodes that took place in order for the query to be resolved. An example of how the nodes are receiving their IDs on the Chord network by hashing their MAC address is shown in the table II.

The experiment that was conducted to test the functionality of this new proposed Chord for WSNs, is the measurement of the lookup query path length between two nodes of the network of N total nodes. For each distinct value of N in the range $[10, 500]$ with the value of N incremented by 5 in each run of the experiment, a total of 10 lookup queries between two randomly selected nodes of the network was executed and the average query path length of these queries was calculated. The combined results of each run of the experiment are displayed on the figure 4. The conclusions of these experiments are that in a network of N nodes, the lookup query needs $O(\log N)$ hops to be resolved, as it was theoretically suggested.

V. Conclusions and Future Work

In the context of this paper, a modified version of the Chord protocol is presented in which we propose an alternative version applicable to the model of Wireless Sensor Networks (WSNs). The main objective of this new approach was to exploit the routing efficiency between the nodes of the Chord protocol which is a decentralized P2P Distributed Hash Table and to bring the more efficient routing functionality in the queries between the nodes of a Wireless Sensor Network. Also, the Chord's support of dynamic networks with nodes joining and leaving make it ideal for WSNs with nodes relying on batteries for power. The modified operations for join, leave or lookup were presented along with their complexity and a LoRa deployment scenario was introduced. Moreover, a simulation of the modified version of the Chord protocol took place, in which nodes are hashed based on their MAC addresses and then, experiments are performed on a test network to measure the lookup query path per total number of nodes on the network. Results are encouraging that the lookup operation follows the theoretical calculated complexity.

Future directions of this work include integration with other technologies, extension of the proposed methodology, experiments among a large set of nodes as well as encryption on the IDs of the sensor nodes so as to increase the security of the overall system. Additionally, a Hybrid P2P system can be developed in which an index server can include all information regarding the nodes and the locations of all resources.

Ultimately, the modified version of Chord could be utilized in energy efficient self-management systems where nodes perform their own actions for battery saving but this requires further investigation.

TABLE II
EXAMPLE OF NODE ID ASSIGNMENT USING SHA-1 TO HASH THE MAC ADDRESS ON A 10-BIT CHORD

Device No.	MAC Address	Node ID
1	7a:60:ed:29:74:e5	27
2	2a:54:27:c4:ec:2e	29
3	f8:59:f1:c3:5a:c7	73
4	d9:38:4e:58:ab:c6	161
5	50:e4:2e:2d:37:a9	382
6	08:b0:08:44:85:60	880

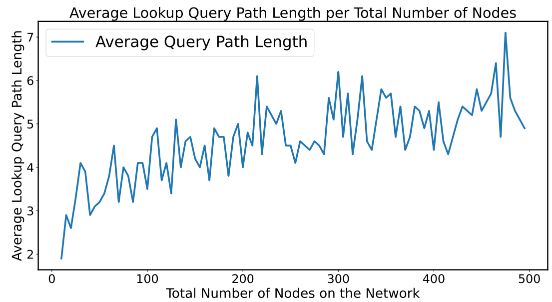


Fig. 4. Combined Simulation Results

References

[1] K. Rose, S. Eldridge, and L. Chapin, "The internet of things: An overview," *The internet society (ISOC)*, vol. 80, pp. 1–50, 2015.

[2] S. Li, L. D. Xu, and S. Zhao, "The internet of things: a survey," *Information systems frontiers*, vol. 17, no. 2, pp. 243–259, 2015.

[3] L. Atzori, A. Iera, and G. Morabito, “The internet of things: A survey,” *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[4] D. Kandris, C. Nakas, D. Vomvas, and G. Koulouras, “Applications of wireless sensor networks: an up-to-date survey,” *Applied System Innovation*, vol. 3, no. 1, p. 14, 2020.

[5] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui, and T. Watteyne, “Understanding the limits of lorawan,” *IEEE Communications magazine*, vol. 55, no. 9, pp. 34–40, 2017.

[6] M. Singh Bali, K. Gupta, K. Kour Bali, and P. K. Singh, “Towards energy efficient NB-IoT: A survey on evaluating its suitability for smart applications,” *Materials Today: Proceedings*, vol. 49, pp. 3227–3234, 2022.

[7] M. I. Hossain and J. I. Markendahl, “Comparison of LPWAN Technologies: Cost Structure and Scalability,” *Wireless Personal Communications*, vol. 121, no. 1, pp. 887–903, 2021.

[8] J. Qin, Z. Li, R. Wang, L. Li, Z. Yu, X. He, and Y. Liu, “Industrial Internet of Learning (IIoL): IIoT based pervasive knowledge network for LPWAN—concept, framework and case studies,” *CCF Transactions on Pervasive Computing and Interaction*, vol. 3, no. 1, pp. 25–39, 2021.

[9] A. Khalifeh, K. A. Aldahdouh, K. A. Darabkh, and W. Al-Sit, “A Survey of 5G Emerging Wireless Technologies Featuring LoRaWAN, Sigfox, NB-IoT and LTE-M,” in *2019 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET)*, pp. 561–566, 2019.

[10] W. Ayoub, A. E. Samhat, F. Nouvel, M. Mroue, and J.-C. Prévotet, “Internet of mobile things: Overview of lorawan, dash7, and nb-iot in lpwans standards and supported mobility,” *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1561–1581, 2018.

[11] L. Oliveira, J. J. Rodrigues, S. A. Kozlov, R. A. Rabêlo, and V. H. C. d. Albuquerque, “Mac layer protocols for internet of things: A survey,” *Future Internet*, vol. 11, no. 1, p. 16, 2019.

[12] B. Shilpa, R. Radha, and P. Movva, “Comparative Analysis of Wireless Communication Technologies for IoT Applications,” in *Artificial Intelligence and Technologies*, pp. 383–394, Springer, 2022.

[13] I. Stoica, R. Morris, D. Liben-Nowell, D. R. Karger, M. F. Kaashoek, F. Dabek, and H. Balakrishnan, “Chord: a scalable peer-to-peer lookup protocol for internet applications,” *IEEE/ACM Transactions on networking*, vol. 11, no. 1, pp. 17–32, 2003.

[14] D. S. Milojicic, V. Kalogeraki, R. Lukose, K. Nagaraja, J. Pruyne, B. Richard, S. Rollins, and Z. Xu, “Peer-to-peer computing,” 2002.

[15] D. Schoder and K. Fischbach, “Peer-to-peer prospects,” *Communications of the ACM*, vol. 46, no. 2, pp. 27–29, 2003.

[16] G. S. Manku, “Routing networks for distributed hash tables,” in *Proceedings of the twenty-second annual symposium on Principles of distributed computing*, pp. 133–142, 2003.

[17] J. Byers, J. Considine, and M. Mitzenmacher, “Simple load balancing for distributed hash tables,” in *International Workshop on Peer-to-Peer Systems*, pp. 80–87, Springer, 2003.

[18] Q. H. Vu, M. Lupu, and B. C. Ooi, *Peer-to-peer computing: Principles and applications*. Springer, 2010.

[19] A. Karras, C. Karras, G. Drakopoulos, D. Tsolis, P. Mylonas, and S. Sioutas, “SAF: A Peer to Peer Iot Lora System for Smart Supply Chain in Agriculture,” in *Artificial Intelligence Applications and Innovations*, (Cham), pp. 41–50, Springer International Publishing, 2022.

[20] S. Sioutas, E. Sakkopoulos, A. Panaretos, D. Tsoumakos, P. Gerolymatos, G. Tzimas, and Y. Manolopoulos, “D-p2p-sim+: A novel distributed framework for p2p protocols performance testing,” *Journal of Systems and Software*, vol. 100, pp. 211–233, 2015.

[21] S. Sioutas, G. Papaloukopoulos, E. Sakkopoulos, K. Tsichlas, and Y. Manolopoulos, “A novel distributed p2p simulator architecture: D-p2p-sim,” in *Proceedings of the 18th ACM conference on Information and knowledge management*, pp. 2069–2070, 2009.

[22] S. Sioutas, A. Panaretos, I. Karydis, and D. Tsoumakos, “Sart: dynamic p2p query processing in sensor networks with probabilistic guarantees,” in *Proceedings of the 27th Annual ACM Symposium on Applied Computing*, pp. 847–852, 2012.

[23] D. Tsolis, S. Sioutas, A. Panaretos, I. Karydis, and K. Oikonomou, “Decentralized digital content exchange and copyright protection via p2p networks,” in *2011 IEEE Symposium on Computers and Communications (ISCC)*, pp. 1056–1061, IEEE, 2011.

[24] D. Tsolis, E. Georgatos, and S. Sioutas, “The use of peer-to-peer networks in copyright protection,” in *2009 13th Panhellenic Conference on Informatics*, pp. 97–101, IEEE, 2009.

[25] S. Sioutas, K. Oikonomou, G. Papaloukopoulos, M. Xenos, and Y. Manolopoulos, “Building an efficient p2p overlay for energy-level queries in sensor networks,” in *Proceedings of the International Conference on Management of Emergent Digital EcoSystems*, pp. 361–368, 2009.

[26] S. A. Sioutas, “Nbdt: An efficient p2p indexing scheme for web service discovery,” *International journal of Web engineering and technology*, vol. 4, no. 1, pp. 95–113, 2008.

[27] S. Sioutas, E. Sakkopoulos, C. Makris, B. Vassiliadis, A. Tsakalidis, and P. Triantafillou, “Dynamic web service discovery architecture based on a novel peer based overlay network,” *Journal of Systems and Software*, vol. 82, no. 5, pp. 809–824, 2009.

[28] G. Fersi, W. Louati, and M. Ben Jemaa, “Distributed hash table-based routing and data management in wireless sensor networks: a survey,” *Wireless networks*, vol. 19, no. 2, pp. 219–236, 2013.

[29] C. Yan, “Distributed hash table based routing algorithm for wireless sensor networks,” in *2014 Fifth International Conference on Intelligent Systems Design and Engineering Applications*, pp. 430–433, IEEE, 2014.

[30] T. Das, P. M. Khilar, and R. R. Swain, “Range-free intersecting chord-based geometric localization scheme for wireless sensor networks,” *International Journal of Communication Systems*, vol. 33, no. 3, p. e4217, 2020.

[31] M. Singh and P. M. Khilar, “A range free geometric technique for localization of wireless sensor network (wsn) based on controlled communication range,” *Wireless Personal Communications*, vol. 94, no. 3, pp. 1359–1385, 2017.

[32] A. Amer, A. Abdel-Hamid, and M. A. El-Nasr, “Chord-enabled key storage and lookup scheme for mobile agent-based hierarchical wsn,” in *International Conference on Security of Information and Communication Networks*, pp. 27–43, Springer, 2013.

[33] J. Yu, W. Liu, and J. Song, “C2wsn: A two-tier chord overlay serving for efficient queries in large-scale wireless sensor networks,” in *15th International Conference on Advanced Computing and Communications (ADCOM 2007)*, pp. 237–242, IEEE, 2007.

[34] L. Cheklat, M. Amad, M. Omar, and A. Boukerram, “Energy efficient physical proximity based chord protocol for data delivery in wsns,” in *2018 International Conference on Applied Smart Systems (ICASS)*, pp. 1–6, IEEE, 2018.

[35] L. Cheklat, M. Amad, M. Omar, and A. Boukerram, “Chearp: Chord-based hierarchical energy-aware routing protocol for wireless sensor networks,” *Computer Science and Information Systems*, vol. 18, no. 3, pp. 813–834, 2021.

[36] M. Ali and K. Langendoen, “A case for peer-to-peer network overlays in sensor networks,” in *International Workshop on Wireless Sensor Network Architecture (WWSNA'07)*, pp. 56–61, 2007.

[37] A. Awad, C. Sommer, R. German, and F. Dressler, “Virtual cord protocol (vcp): A flexible dht-like routing service for sensor networks,” in *2008 5th IEEE International Conference on Mobile Ad Hoc and Sensor Systems*, pp. 133–142, IEEE, 2008.

[38] N. Bhalaji, S. J. Prasanna, and N. Parthiban, “Performance analysis of creido enhanced chord overlay protocol for wireless sensor networks,” in *Proceedings of the International Conference on Data Engineering and Communication Technology*, pp. 489–499, Springer, 2017.

[39] M. D'Angelo and M. Caporuscio, “Sa-chord: A self-adaptive p2p overlay network,” in *2018 IEEE 3rd International Workshops on Foundations and Applications of Self* Systems (FAS* W)*, pp. 118–123, IEEE, 2018.

[40] A. Rowstron and P. Druschel, “Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems,” in *IFIP/ACM International Conference on Distributed Systems Platforms and Open Distributed Processing*, pp. 329–350, Springer, 2001.