

Abdel Ghafar, Ahmed Ismail; Vazquez Castro, Ágeles; Essam Khedr, Mohamed

**Conference Paper**

## Multidimensional Self-Organizing Chord-Based Networking for Internet of Things

2nd Europe - Middle East - North African Regional Conference of the International Telecommunications Society (ITS): "Leveraging Technologies For Growth", Aswan, Egypt, 18th-21st February, 2019

**Provided in Cooperation with:**

International Telecommunications Society (ITS)

*Suggested Citation:* Abdel Ghafar, Ahmed Ismail; Vazquez Castro, Ágeles; Essam Khedr, Mohamed (2019) : Multidimensional Self-Organizing Chord-Based Networking for Internet of Things, 2nd Europe - Middle East - North African Regional Conference of the International Telecommunications Society (ITS): "Leveraging Technologies For Growth", Aswan, Egypt, 18th-21st February, 2019, International Telecommunications Society (ITS), Calgary

This Version is available at:

<https://hdl.handle.net/10419/201736>

**Standard-Nutzungsbedingungen:**

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

**Terms of use:**

*Documents in EconStor may be saved and copied for your personal and scholarly purposes.*

*You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.*

*If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.*

# Multidimensional Self-Organizing Chord-Based Networking for Internet of Things

Ahmed Ismail Abdel Ghafar<sup>§</sup>, Ángeles Vazquez Castro<sup>§</sup>, Mohamed Essam Khedr<sup>±</sup>

<sup>§</sup>Autonomous University of Barcelona (UAB), Spain

<sup>±</sup>Electronics and Communication Department, Arab Academy for Science and Technology and Maritime Transport (AASTMT), Egypt

**Abstract** - IoT is a coin term recently used in ICT research and industrial community to express the involvement of devices of different capabilities and functionalities in the daily activities of people and organizations. With the enormous amount of data generated by highly dynamic users, the problem of storing, looking up, validating and manipulating data becomes crucial for the success of future networks. A multidimensional chord peer-peer network as extension to the successful chord technology is proposed to cope with the dynamism of IoT networks. Novel approaches have been developed to tackle the high frequency of nodes joining and leaving/failure the network and to deal with Big data, similarity of data, filtering and Geo data.

**Keywords:** Internet of Things, peer to peer networks, multidimensional chord networks, distributed resource sharing

## 1 Introduction

The Internet of Things (IoT) is a new paradigm in the field of networking that consists of massive number of sensors and computing devices with capability of interacting with one another in autonomously. In addition, IoT will enrich Information and Communication Technology (ICT) research and industrial community through the involvement of devices of different capabilities and functionalities in the daily activities of people and organizations. The true value of the IoT can be fully realized when connected devices are able to communicate with one another and integrate with vendor-managed inventory systems, customer support systems, business intelligence applications, and business analytics [1].

Mobile phones, wearable devices, sensors, appliances and automobiles are examples of such devices in IoT environment while users' tasks, transportation systems, health care systems, smart digital cities and educational systems are examples of applications where IoT will be used.

To achieve such a dynamic and scalable network, we need to have successful interoperation between all systems such as: ID, Meta information, Security control, service discovery, relationship management and service composition [2].

With such vast highly dynamic environment, challenges rise on methods and techniques that will be used to discover, connect, organize such devices from one side and mechanisms to collect, aggregate, filter, process, store and retrieve generated data from others.

Nowadays, data that we produce through our devices are processed and stored by centralized Internet companies, even if data are distributed among different servers of the company, there is a single point of control and access to them. This approach of entrusting people's data to centralized Internet companies has already proved to constitute a threat to people's privacy. Such threat is even more concrete in the context of the IoT, since billions of devices will continuously collect fine-grained personal information [3]. Starting from these risks of the current data management model, our work will consider decentralized Peer to Peer (P2P) network where IoT devices store data in a collaborative manner (instead of entrusting them to centralized companies). We focus on designing an overlay management architecture based on chord algorithm, so the proposed solution is an extension to the successful chord technology to cope with the dynamism of IoT networks.

The chord algorithm is a peer-to-peer lookup system. Given a key (data item), it maps the key onto a node (peer) using a consistent hashing to assign keys into nodes. Generally, it solves of locating key in a collection of distributed nodes. Advantages of using chord are:

- Load balance: due to the usage of distributed hash function, keys are spread evenly over nodes.
- Decentralization: chord is fully distributed, no node more important than other thus improving robustness.
- Scalability: logarithmic growth of lookup costs with number of nodes in network, even very large systems are feasible
- Availability: chord automatically adjusts its internal tables to ensure that the node responsible for a key can always be found.

However, chord networks have the limitations of not being able to distinguish type of data stored, no control on process of modifying data that is stored, lookup delay increases significantly with the growth of the chord and finally robustness may be jeopardized in case of simultaneous failure and/or add-join of nodes.

To overcome such limitations, the proposed solution consists of multi-dimension chords. Each dimension represents a type of data collected from sensors that provide this kind of data. The

architecture of proposed multi-dimension chord consists of a main chord network of edge devices that branch from them sub-chords of specialized nodes. We also bring new commands (operations) to standard chord networks such as: remove shared node, merge shared chords, split shared chords, leave from shared node to single node, join shared chord, sleep single node, sleep shared node, awake single node and awake shared node.

This paper focuses on the design principles of the proposed novel chord network-multidimensional chord-based networking for IoT devices based on well-known chord peer to peer network. The remainder of this paper is organized as follows: in section 2, the literature review and related work is summarized; in section 3, we describe the system model; in section 4, we present the system analysis; in section 5, we discuss the simulation results; finally, we conclude this paper and the future work in section 6.

## 2 Related Work

There are a lot of research that has been carried out for the network architectures and flow of information of IoT networks. These works are classified according to their approaches.

There is no single consensus on architecture for IoT, which is agreed universally. Different architectures have been proposed by different researchers [2]. Three- and Five-Layer Architectures are the most basic architectures as shown in figure 1. It was introduced in the early stages of research in this area. The three layers consists of the perception, network, and application layers.

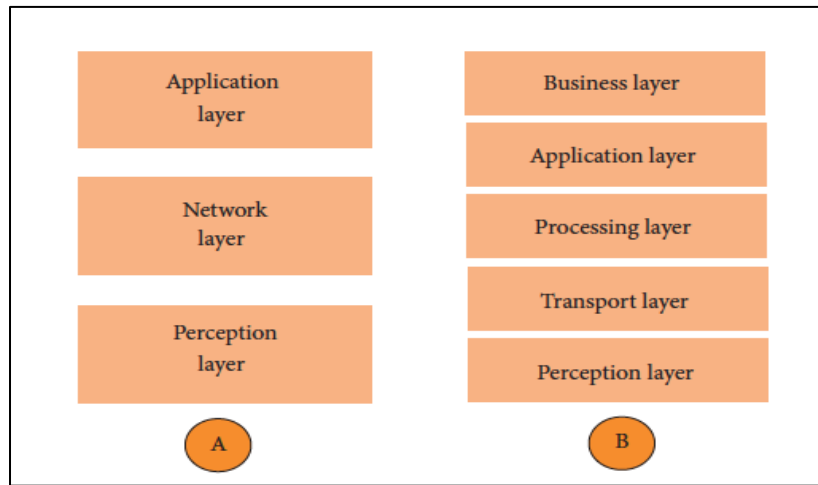


Figure 1: IoT architecture

(i) The perception layer is the physical layer, which has sensors for sensing and gathering information about the environment. It senses some physical parameters or identifies other smart objects in the environment.

(ii) The network layer is responsible for connecting together smart things, network devices, and servers. Its features are also used for transmitting and processing sensor data.

(iii) The application layer is responsible for delivering application specific services to the user. It defines various applications in which the Internet of Things can be deployed, for example, smart homes, smart cities, and smart health. The three-layer architecture defines the main idea of the Internet of Things, but it is not sufficient for research on IoT because research often focuses on finer aspects of the Internet of Things. That is why, we have many more layered architectures proposed in the literature. One is the five-layer architecture, which additionally includes the processing and business layers. The five layers are perception, transport, processing, application, and business layers (see Figure 1). The role of the perception and application layers is the same as the architecture with three layers. Below is the function of the remaining three layers.

(i) The transport layer transfers the sensor data from the perception layer to the processing layer and vice versa through networks such as 3G, LAN, Bluetooth, RFID, and NFC.

(ii) The processing layer is also known as the middleware layer. It stores, analyzes, and processes huge amounts of data that comes from the transport layer. It can manage and provide a diverse set of services to the lower layers. It employs many technologies such as databases, cloud computing, and big data processing modules.

(iii) The business layer manages the whole IoT system, including applications, business and profit models and users' privacy. The business layer is out of the scope of this paper. Hence, we do not discuss it further.

Smart Cities in [4] uses Information Technology (IT) to integrate and provide better services to its dwellers while ensuring efficient and optimal utilization of available resources. They proposed a security framework that integrates the BlockChain technology with smart devices to provide a secured communication platform in a smart city. But BlockChain are computationally expensive and involve high bandwidth, which aren't suitable for most IoT networks[5], they proposed a lightweight BlockChained-based architecture for IoT that virtually eliminates the overheads of classic BlockChains, while maintaining most of its security and privacy benefits.

BlockChains allow us to have a distributed peer-to-peer network where non-trusting members can interact with each other without a trusted intermediary [6].

In [7], they were determining the node that stores a data item in a distributed network is an important and challenging problem. Based on the experience using the Chord system for a peer-to-peer file sharing application and the results from theoretical analysis, simulation studies with up to 10,000 nodes, and experiments, it can be concluded that the Chord protocol and system is a valuable component for many decentralized, large-scale distributed applications.

In [8], they focused on designing an overlay management architecture based on Chord algorithm to manage things in heterogeneous space. The architecture is a collection of multiple peer Chord rings, and thus each of these rings is considered as a smart context. They also brought forward operation mechanisms for the architecture such as thing identification, key insert/remove, key lookup and node joining/leave. The architecture was tested by simulating a large-scale environment with a large number of nodes and sensors to evaluate performance and operation costs of the entire system with various topologies and properties. The achieved results prove that the proposal had the operability and feasibility to apply to practice.

### **3 Multidimensional scalable chord architecture**

There has been a remarkable growth of digital devices, such as sensors, actuators, smart phones and smart appliances due to the possibility to interconnect all devices and create communications between them. However, this will require an effective and efficient way of gathering information from such massive number of devices. For example, public transportation systems will need to store and access real-time location information, availability of parking spaces in city, traffic jams, and other data like weather conditions, air and noise pollution status, water contamination, energy consumption, etc.

In this direction, we consider the environment as an integration of multidimensional logical Chord rings, which could be represented as a one main chord. Each node on the ring is considered as an IoT gateway and plays the role of transmitter and connector to sensors. Then, another sub-chords are deviated from this main chord based on a certain criteria which is under evaluation and still

needs further tuning based on the simulation results. Sub-Chords are divided logically based on type of information in each.

### **3.1 Chord Algorithm**

Chord is widely known as one of popular DHT algorithms to reduce significantly the number of hops for looking up and updating in peer-to-peer (P2P) network. By using consistent hashing, Chord organizes nodes into a ring that can cover the huge number of nodes (e.g. SHA-114 provides 160-bit keys). Each node maintains routing information of other  $O(\log N)$  nodes. Thus, only  $O(\log N)$  hops are required for each lookup request and  $O(\log^2 N)$  hops for updating when a node joins or leaves the ring. Due to its advantage features such as load balance, decentralization, scalability, availability and flexible naming, Chord is applied to design our logical management system in order to meet requirements.

### **3.2 Multidimensional chord architecture**

The multidimensional Chord rings architecture is divided into three components, namely ring, node, and sensor, which are defined as follows:

- Ring - is logical management network for a single IoT context. Ring uses Chord as a base method to organize and discover things.
- Node - is a peer object located on the ring. Node corresponds with a gateway device in fact. We divide nodes into two types:
  - Single node - is a node that is participating in only one ring.
  - Shared node - is a node that is simultaneously participating in at least two rings.
- Sensor - is an object representing sensors in practice. In our multi-ring architecture, sensors cannot join directly on rings like nodes (i.e gateway) because they manifest characteristics of real devices, which are limited by capabilities of communication, computing and storage. Therefore, sensors must connect to a certain gateway. Connection information between a sensor and its gateway is distributed on ring nodes.

Particularly, on each ring, organization of things (i.e. nodes and sensors) are performed based on traditional Chord. Figure 2 illustrates an example for obtaining multidimensional chord.

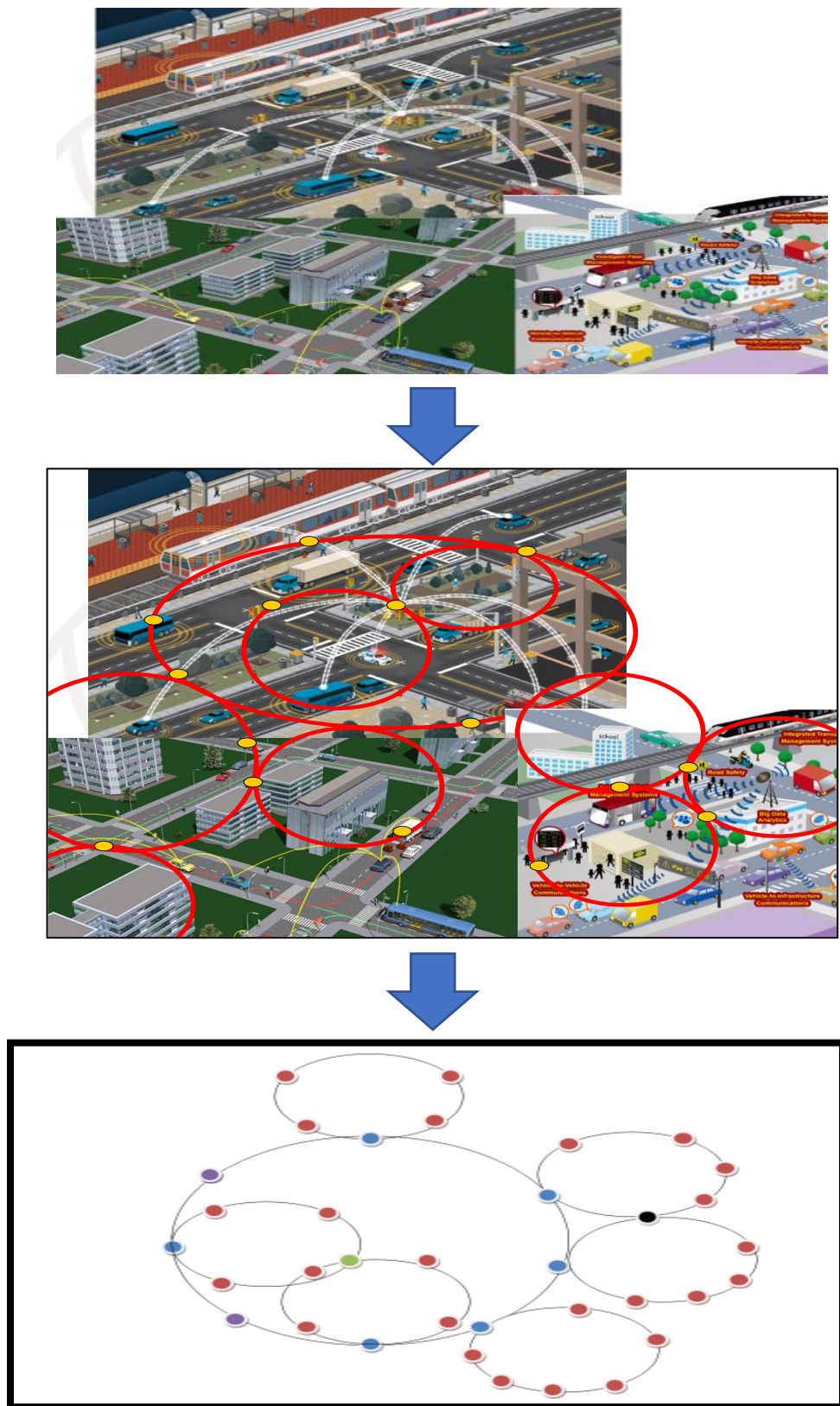


Figure 2: Multidimensional Chord example



## 4 System analysis

### 4.1 Identification

Consistent hash function assigns each node and key an m-bit identifier, SHA-1 is used as a base hash function. A node's identifier is defined by two values:

- a- Hashing the node's IP address and port number to define its value in a Subchord. A key identifier is produced by hashing the key.

$$\text{ID}(\text{node}) = \text{hash}(\text{IP}, \text{Port})$$

- b- Hashing the type of data to be stored and retrieved in this Subchord.

$$\text{ID}(\text{type}) = \text{hash}(\text{Type})$$

In an m-bit identifier space, there are  $2^m$  identifiers. Identifiers are ordered on an identifier circle modulo  $2^m$ , the identifier ring is called Subchord ring. Key k is assigned to the first node whose identifier is equal to or follows (the identifier of) k in the identifier space. This node is the successor node of key k, denoted by  $\text{successor}(k)$ .

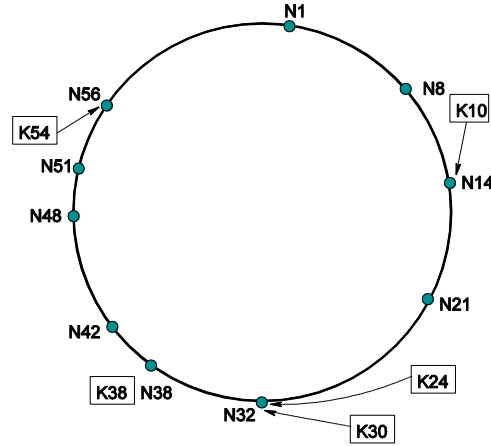


Figure 3: Sub-chord of data type "weather"

### 4.2 Key Insert and Remove

Two operations are needed for a key to be inserted or removed from the multidimensional chord network. The first operation is to lookup the Subchord responsible for this kind of data type and

the second operation is to find the node in such Subchord that will store in it or remove the data from it.

The first operation is carried using the Subchord table which is very similar to the finger table in concept except that the entries in the Subchord table are the heads of each Subchord implemented in the system. Thus is it like a highway where the search will be made on the level of the Subchord heads only.

The second operation is carried using the finger tables of the nodes in the Subchord that has been chosen from the first operation. This is a standard operation similar to the one used in single chord model. The two operations are illustrated in figure 4.

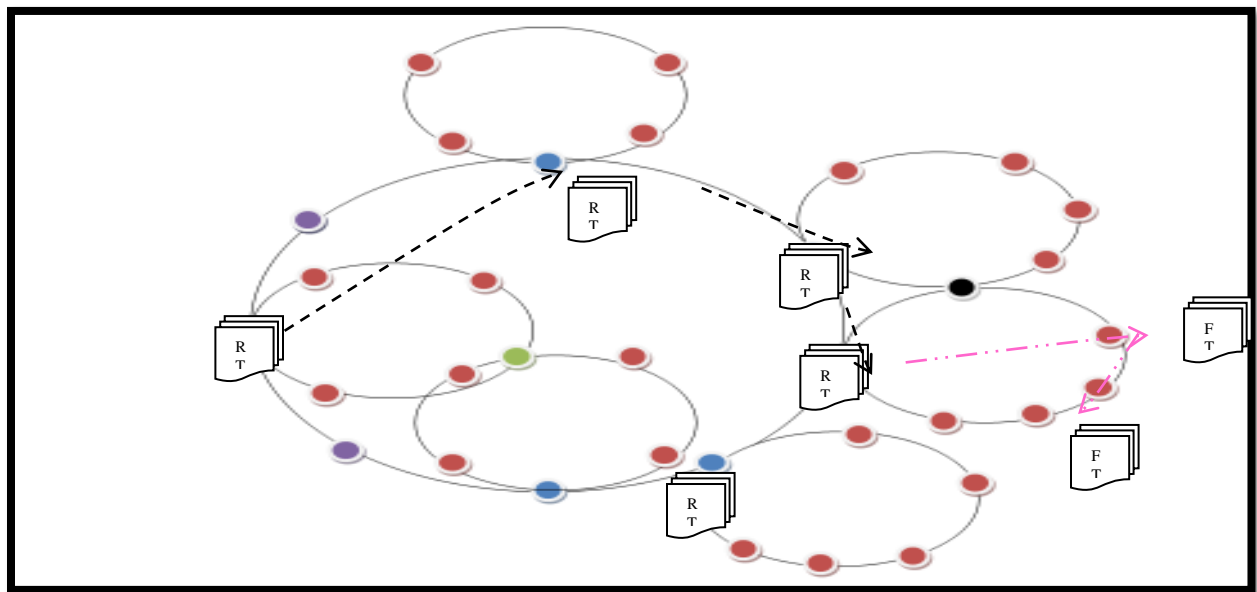


Figure 4: Key Insert and Remove

### 4.3 Node Join and Leave

Another task that is required in a multidimensional chord network is the joining of new nodes in a Subchord or leaving of existing nodes from a Subchord. The two operations require first to adjust the keys stored in the nodes of a Subchord and to update the finger and Subchord tables accordingly. The two operations are similar to the standard join/leave operations in a single chord model but with the addition of updating the Subchord tables of the head nodes in the multidimensional chord network.

## 5 Simulation Results

A validation of the multidimensional chord network was carried using a discrete simulator software. We investigated the time responsiveness of the model with respect to generating a Subchord, adding keys to Subchord, joining and leaving a Subchord.

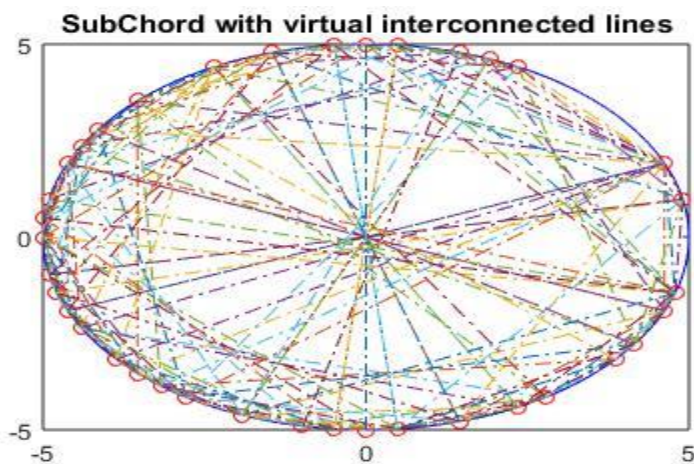


Figure 5: Subchord with interconnected lines from Finger table.

Figure 5 shows the Subchord generated by proposed system for group of nodes that are common in the type of data that they produce. The interconnected lines between the nodes are virtual connections that show each node is connected to which other nodes using the finger table model. As the figure shows, all nodes can be reached from any single node in the Subchord.

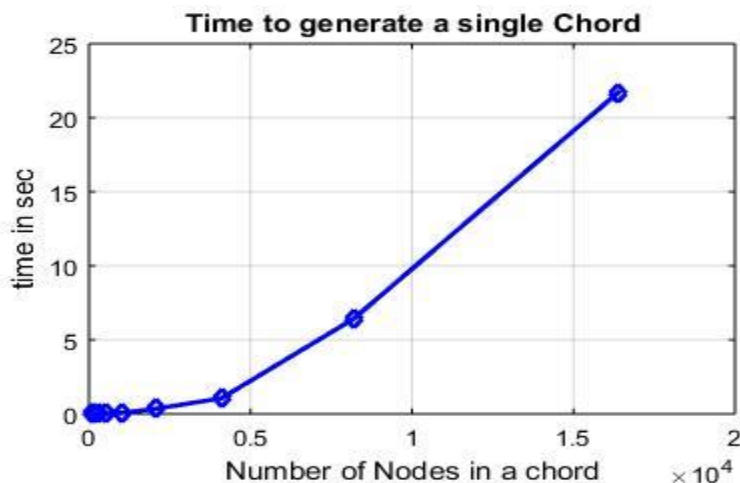


Figure 6: Time for generating s Subchord

Figure 6 shows the time needed to generate a Subchord along with the number of nodes in the Subchord.

As shown in figure the time taken is quazi-linear with the number of the nodes in the Subchord. This proves that as the number of nodes exceeds certain value, the time to generate the Subchord may be inappropriate and thus a split of the Subchord is needed into two smaller Subchords to maintain the time of deploying a Subchord.

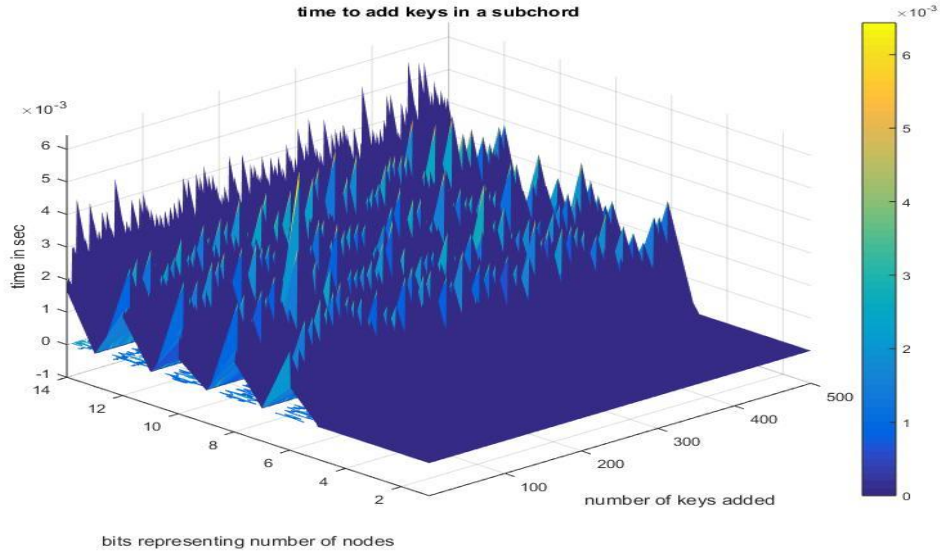


Figure 7: Time to add keys to a Subchord

Figure 7 shows the time needed to add a new key in a Subchord. The figure shows the time of adding 500 keys in Subchords that have nodes ranging from 64 node up to 6000 nodes. The figure illustrates the applicability of the proposed model to accommodate keys with no significant change in time needed to do such operation.

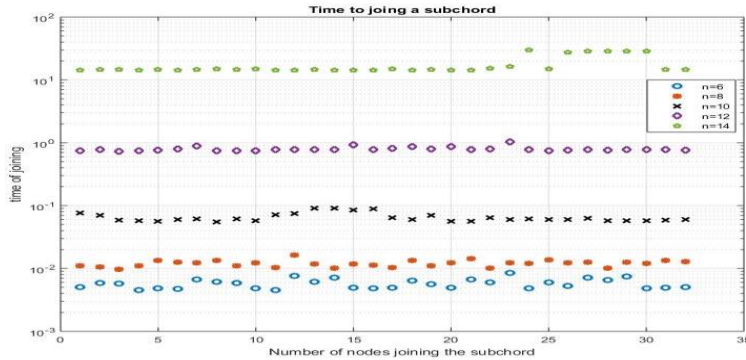


Figure 8: Time needed to join Subchord

The time needed for nodes to join a Subchord is illustrated in figure 8 where  $n$  is the number of bits used in the hashing algorithm to represent the number nodes in the chord which is equal to  $2^n$ . As shown in the figure the required time to join a Subchord of certain capacity is nearly constant. But as the capacity increases the time required to join the Subchord becomes large which signifies again the importance to have a dynamic multidimensional chord network instead of a single chord network. This is further proved in figure 9 that shows the time required to leave a Subchord in the system.

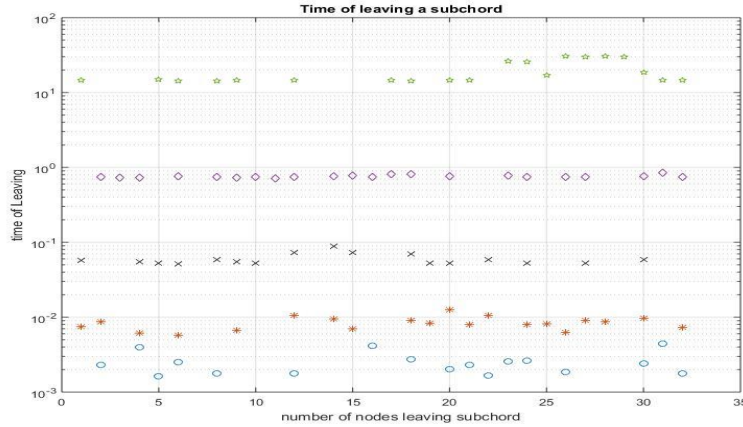


Figure 9: Time needed to leave Subchord

## 6 Conclusion and Future work

In this paper we presented a multidimensional chord based network for IOT systems. The proposed model extends the chord peer to peer network into multiple chords that are emerged from a main chord and according to their specific tasks. The identification of each node in the system is a tuple of two entries. The first entry is a hashing of the IOT data type and the second entry is hashing of the IP and port addresses. The proposed system suggested new commands to cope with the new architecture especially adding new chords, defining shared nodes and shared chord. We presented in the simulation a validation of the proposed model and how the process works for adding, removing nodes as well as manipulating data. As a future work, we intend to make a thorough analysis of the performance of the multidimensional chord based network to test its scalability and fault tolerance capabilities.

## References

1. Lee, I. and Lee, K. (2015). The Internet of Things (IoT): Applications, investments, and challenges for enterprises. *Business Horizons*, 58(4), pp.431-440.
2. Sethi, P. and Sarangi, S. (2017). Internet of Things: Architectures, Protocols, and Applications. *Journal of Electrical and Computer Engineering*, 2017, pp.1-25.
3. Conoscenti, M., Vetro, A. & Martin, J.C.D., 2017. Peer to Peer for Privacy and Decentralization in the Internet of Things. 2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C).
4. Biswas, K. & Muthukkumarasamy, V., Securing Smart Cities Using Blockchain Technology. 2016 IEEE 18th International Conference on High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems, 178, pp.1392–1393.
5. Dorri, A., Kanhere, S.S. & Jurdak, R., 2017. Towards an Optimized Blockchain for IoT. Proceedings of the Second International Conference on Internet-of-Things Design and Implementation - IoTDI 17.
6. Christidis, K. & Devetsikiotis, M., 2016. Blockchains and Smart Contracts for the Internet of Things. *IEEE Access*, 4, pp.2292–2303.
7. Stoica, I. et al., 2001. Chord. *ACM SIGCOMM Computer Communication Review*, 31(4), pp.149–160.
8. Nguyen, B.M. et al., 2017. Multiple Peer Chord Rings Approach for Device Discovery in IoT Environment. *Procedia Computer Science*, 110, pp.125–134.