Performance Evaluation of Distributed Networks for Internet of Things

Nithya Balasubramanian, Deshpande Rutvik, Raj Sureja and Gollapalli Sai Pavan

Department of Computer Science and Engineering, National Institute of Technology, Tiruchirappalli, 620015, Tamil Nadu, India

Abstract

The growing prevalence of Internet of Things (IoT) has placed a strain on the conventional cloud based network architecture. Currently, there is active research on exploring distributed approaches towards IoT in order to solve the scalability issues, among others. However, the majority of this research focuses on solving implementation issues that occur for specific approaches, and there is a lack of research that focuses on comparing approaches in order to find out which would be better to develop and implement. In order to expand upon the lack of information, this paper compares the performance of three approaches to distributed IoT networking: a consensus based approach, an approach based on Information Centric Networking (ICN), and a blockchain network based approach. A simulation system is built in order to perform this comparison in a wide range of scenarios, such as large IoT networks, systems with high user utilization, light or heavy applications, and network conditions. These three models are compared using data generated through simulations of IoT workloads, by looking at how responsive the system is to user requests, request throughput, and the load placed on IoT devices in the network. From the results, it is inferred that an ICN based approach gives the better overall performance while keeping performance costs in check, but requires at least one device in the network that can handle high loads. A consensus based approach showed poor scalability for larger network sizes. Using a blockchain approach provided the absolute fastest response time and throughput, but at the cost of much higher device loads and power consumption.

Keywords 1

Internet of Things, Distributed Network, Consensus, Information Centric Networking, Blockchain, Evaluation

1. Introduction

Over the years, the evolution of technology has resulted in decreased human interaction. From automating manufacturing machines to customer service, computers have become ubiquitous. IoT aims to have a similar effect on a domestic level by streamlining interaction with personal appliances through a simple application [1]. Several technological systems such as real-time analytics, embedded systems and machine learning merge together to form the overarching system called as IoT. The structure of an IoT system can vary from one implementation to another but the core characteristic remains the same - minimal human interaction. Every device in the network is a node. Some of these nodes collect data through sensors. The data is sent to either a central server or to the cloud to be processed. Machine learning and big data concepts can be used to analyze collected data. This improved connectivity has enabled physical devices to be controlled wirelessly leading to a significant degree of automation across several industries. The industry-wide phenomenon has led to better efficiency overall.

WCES-2021: Workshop on Control and Embedded Systems, May 01, 2021, Chennai, India.
EMAIL: nithya@nitt.edu (Nithya Balasubramanian)
ORCID: 0000-0002-5698-3814 (Nithya Balasubramanian)

As with any technological advancement, IoT brings a variety of issues. Some of the major issues pertain to scalability, privacy and security, safety etc. Furthermore, the safety risks are involved with sensitive devices such as cameras and thermostats are worrisome. The impact of software bugs can be significant and could have serious consequences. From a social perspective, people need more time to adapt to an automated world. Then there is the problem of scalability. With the rate at which new devices are manufactured every year, it becomes increasingly difficult to support communication among them. This is seen especially in the case of centralized systems where the central device becomes a bottleneck. There are things to be kept in mind regarding scalability such as the techniques that are used for achieving it and the challenges that using them would bring. Applications have been developed in industries ranging from healthcare to transportation, and in recent years the integration of objects, sensors and cloud computing [2]. However, the development of these technologies is still in its infancy and many challenges lie ahead. Research continues on standardization, security, infrastructure, interfaces and communication protocols, with several competing implementations and organizations.

This paper aims to investigate and compare different approaches to the distributed IoT in a smart home environment in order to understand which approach performs better, and should be developed further. Three approaches to the distributed IoT networking are considered: a consensus based approach that involves true role homogeneity in the network[3], an approach based on Information Centric Networking(ICN) where the focus is on the location of data rather than on devices[4], and a blockchain approach that utilizes blockchain networking in order to leverage its advantages[5]. These three models are chosen as they provide a broad spectrum of potential solutions to the IoT scalability problem, and have been studied for this purpose in prior research. These models will be compared in a wide range of scenarios through simulation, such as large network sizes, differing network conditions, light or heavy applications, and various levels of utilization, in order to provide a comprehensive picture of their performance and effectiveness in solving IoT's current scalability issues.

The rest of the paper is organized as follows: Section 2 elaborates on three networking models along with their application. Moreover, the precise summary about these models are tabulated. The proposed analysis of these models is discussed in Section 3 with the process flows. Section 4 presents the simulation model followed by the detailed analysis and comparison of these models under device count and differing network structures. Finally, conclusion of the proposed analysis is given in Section 5.

2. Related Work

Existing solutions have focused on different technical approaches that include publish/subscribe models, service-oriented architectures, semantic models and context-awareness. This section analyses a few prominent models currently in use in diverse scenarios. Information Centric Networking model and its applications in IoT are discussed. Then, literature regarding Consensus algorithms is reviewed. Finally, various Blockchain based networking models are discussed.

2.1. Information Centric Networking

Information-Centric Networking (ICN) has been proposed as a new distributed paradigm which is quite different from the current host-centric paradigm. It makes the content names persistent and independent of the location. It also provides security and enables easy access of data along with multicast abilities. This makes it an ideal platform to build IoT based solutions [6]. ICN is more concerned with the data itself rather than which node in the network it came from. It treats the data as the main entity as it attaches names to data instead of location. The unique naming of the information ensures that information can be located from any part of the network. It is possible to send data back from a route different from the one used to find it. Quite a few ICN architectures have been proposed in the last 2 decades. Avoiding DNS lookups and instead using the name of the object to find a route is first proposed by the TRIAD.

Later, the Data-Oriented Network Architecture (DONA) made improvements by including security and persistence in the architecture. The Content-Centric Network (CCN) has communication based on content request messages (or "Interests") and content return messages (or "Content Objects"). Scalable and Adaptive Internet Solutions (SAIL) builds upon the CCN by incorporating additional services and is flexible enough to be implemented over various routing technologies. Most IoT communication and application patterns such as content retrieval and mobile content updates, inherently follow the ICN concept. A request and response based communication model has proven to be highly effective for a smart home scenario. An open, secure and flexible platform based on IoT and cloud computing is proposed in [7] for medical purposes. ICN-IoT middleware [8] is developed to support different IoT application by combining both NDN and Mobility First architectures. In [9], the initial design of a CCN based homenet is presented along with the aspects of naming, node and service discovery with a comparison against an IPv6 architecture. Keeping the general functions of IoT in mind, [10] shows how ICN concepts can take over easily. A use case of IoT in Ambient Assisted Living (AAL) is analysed in [11] by using ICN features for interaction between dissimilar devices and services. The summary of these models is given in Table 1. From the aforementioned discussion, ICN seems to be one of the easiest models to implement for distributed networks. By limiting the dependence on location, it opens doors for a variety of methods which can suit particular use cases.

Table 1: Summary of ICN based approaches

S.No	Ref No	Proposed Approach	Inference
1	[7]	Implements ICN features in a cloud environment which provides flexibility and security	ICN architectures are easy to integrate over cloud platforms and can be utilized for certain use cases
2	[8]	Combines 2 ICN architectures and adds middleware to access and change the information for use in controlling IoT.	Functions like service discovery and naming service are implemented using ICN features. This shows the capability of ICN architecture
3	[9]	Highlights benefits of ICN by comparing implementations of features by ICN and IP for a homenet	CCN's advantages of efficient data retrieval, naming, scalability and security are realized
4	[10]	Discusses how ICN architecture can be used to achieve goals of IoT	Characteristics of ICN are shown as capable of being a part of a working solution for different needs of a network like security and routing
5	[11]	IoT architecture designed for AAL using ICN concepts	Communication among heterogeneous devices and services is possible using ICN hence proving its versatility

2.2. Consensus Model

In a distributed network, it is difficult to process the information without the central base station. Nodes work cooperatively to reach a particular decision. Consensus algorithms help to achieve that. The consensus algorithms play a key role in Blockchain based distributed computing models. They can influence the design and effectiveness of the network. They also can improve the fault tolerance of a network and this capability can vary depending on the consensus model used. There exist quite a few consensus mechanisms for distributed systems. Proof of Work (PoW) [12] was the first such algorithm and it was used for Bitcoin. There are also randomized solutions to this problem like the Ben-Or's

consensus algorithm and Rabin's consensus algorithm. These can be classified into Monte Carlo and Las Vegas consensus algorithms. Other categories include leader based, leader-free and deterministic algorithms. The widely used consensus algorithm to reach a decision in a distributed environment is the Gossip algorithm [13]. In Gossip algorithm, pairs of nodes are chosen randomly to exchange information and update their values. Compared with other routing algorithms, it does not need any route discovery and route maintenance. It is easy to implement. But the random information exchange creates more overhead in the network and it takes more time to reach consensus.

In [14], the relevance of Byzantine fault tolerance (BFT) algorithms with respect to blockchain is studied. A solution which is a hybrid of BFT and Nakamoto algorithm is talked about as well. A randomised leader-free consensus algorithm is presented in [15]. This helps to transform the multi value Byzantine consensus problem to binary Byzantine consensus problem. Binary instances of consensus are run in parallel by which a value is decided in constant time. The optimality of proposed design is proven theoretically.

The above mentioned models are summarized in Table 2 and it is inferred that the way of arriving at a consensus can greatly affect the functioning of the network and. The primary implementation of these protocols can be found in relation to Blockchain. Each can perform better than the rest depending on the use case.

Table 2: Summary of Consensus based approaches

S.No	Ref No	Proposed Approach	Inference
1	[12]	Used for permission less public distributed ledger, uses cryptographic puzzles	Requires significant amount of computational resources, unfair advantage to better machines
2	[13]	Used for permissioned private distributed ledger, can tolerate faults and attacks	Suitable for high throughput applications, is very scalable and efficient
3	[14]	Voting mechanism, defined in a consortium model	Shown to be more power efficient than Proof of Work
4	[15]	Uses trusted subnetworks in which protocol is periodically executed	It is quite slow but new transactions are validated

2.3. Blockchain Model

The Blockchain technology first came to prominence in early 2009, through the cryptocurrency Bitcoin. The blockchain is a distributed database maintaining a growing list of blocks. Each block has a timestamp and a link to the previous block. Once data is entered into a block, it cannot be tampered. This is the reason behind using blockchains to record transactions between 2 entities. Users are known by a public key and they broadcast generated transactions across the network. The users push these transactions into the blocks. After the block is full, a mining process is used to attach the block to a Blockchain. A blockchain router based method enables multiple blockchains to communicate with each other. The architecture of the approach consists of four participants; validator, surveillant, nominator and connector [16]. Conceptually, it has been inspired from the routing model of the internet. Routers and terminal devices make up the typical routing environment. In this, the blockchain systems, such as Ethereum etc, correspond to the terminal equipment in the network, which is called “sub-chain”. A sub-chain has the ability to send and receive messages from a router or another sub-chain, however cannot communicate with each other directly. The router can communicate with other blockchain routers or a sub-chain. The exchange of information between the sub-chains and routers ensures smooth operation.

Cross communication between blockchain pairs, called Interchain, was put forward [17]. It contains interchain nodes, validating nodes, gateway nodes sub-chain and Inter-Chain. This model connects

separate blockchains. The solution [18] involves a satellite chain to meet the industrial standards. In this, sub-chains are independent and run their own different consensus algorithms. A regulator oversees the entire network using smart contracts. Heterogeneous consensus algorithms are supported by satellite chains to run in parallel in different sub chains. In order to gain interoperability among heterogeneous blockchains, a solution is proposed [19] using smart contract that helps to share data amongst independent heterogeneous blockchains. Cryptocurrency and token based blockchains benefit from the Blocknet [20] protocol. It provides inter-blockchain services like decentralized exchange. Most cryptocurrencies are supported by it. A blockchain-based software platform called Ethereum helps to build and run smart contracts for distributed applications. The internal accounts are represented by the smart contract and these interact with external accounts, which are the users on the network. The state transitions change the balances in the accounts. The balances and various other data of the accounts make up the state at that point in time. A Merkle Patricia tree is used by accounts to maintain an encoded version of the state.

Table 3: Summary of Block Chain based approaches

S.No	Ref No	Proposed Approach	Inference
1	[17]	Uses subchains with validating and gateway nodes	It enables cross Communication between blockchains
2	[18]	A regulator used with independent Subchains	Meets industrial standards with a satellite chain design
3	[19]	Blockchains are heterogenous and independent	Uses smart contracts to enable Interoperable heterogeneous Blockchains
4	[20]	Uses multiple chain communication layer	Multiple architectures for reliable asset transfer across the network

From the above Tables 1,2 and 3, it is inferred that there exists a wide body of research that performs exploratory work both on distributed networking technologies, and how these can be applied in the case of IoT in order to address growing issues with scalability, privacy etc. Most of this research is, however, narrow and focused in scope, addressing questions on how to address specific challenges faced by distributed networking and processing. As such, there is very little investigation into the comparison between these different approaches, which leaves current research scope rather broad. This paper aims to address this by comparing the three different models explored above in order to provide a better understanding of which model works best in what scenario, such as for larger IoT networks, for very low power devices, for networks that would experience high user loads, and so on. By evaluating the performance of these devices in an empirical manner, a perspective can be provided on the benefits and costs involved with each of the models, which would help the decision making process on which approach should be further developed and implemented.

3. Proposed Analysis of Distributed IoT Communication Models

Looking at different approaches to solve the IoT scalability problem through decentralization of workloads, this paper focuses on a breadth-first approach in order to provide preliminary comparison among major alternatives. Three models are evaluated here: a consensus-based approach to truly distributed IoT, a hybridized approach based on a new paradigm in networking called Information Centric Networking (ICN), and a blockchain focused solution that handles distributed communication through established blockchain protocols. Although most final communication in a distributed IoT network should occur directly between user and device, the networking models are responsible for many of the other data flows needed for the entire system to work, such as inter-device communication, device discovery and trust, security, and user authentication, among others. In order to provide a baseline for comparing the efficiency of the network models in addressing these data flows, the models are

compared with an “ideal” distributed IoT network that requires no overhead, and only uses direct P2P communication.

The overall architecture of the proposed analysis involves two main components: a simulator and a data processor. Various input variables, such as the number of devices in the IoT system and the intensity of the application are used to set up the simulation environment and evaluate the response of each model to the changes in these input variables. This, along with the actual communication model (i.e. consensus/ICN/blockchain), is used by the simulator to simulate user and system workloads on the IoT network. The simulator logs all the events and transactions that have occurred during the course of a single simulation run, and saves them to the file. The data processor then takes the logs, and processes the data in order to generate readable output in the form of three output variables such as throughput, response time, and device load. By varying each input variable independently for all models and combining the outputs, graphs can be generated that provide a base for comparing all the models.

3.1. Process Flow in Consensus based approach

Consensus models focus on considering information from every single device in the network in order to make a decision, such as when authenticating users. This means that even for the simple requests, every other node has to be polled for opinion. The response from every node is then evaluated with a consensus algorithm such as BFT, and the user request is finally processed. Models based on consensus hence are homogeneous, and every device in the network has an equal responsibility.

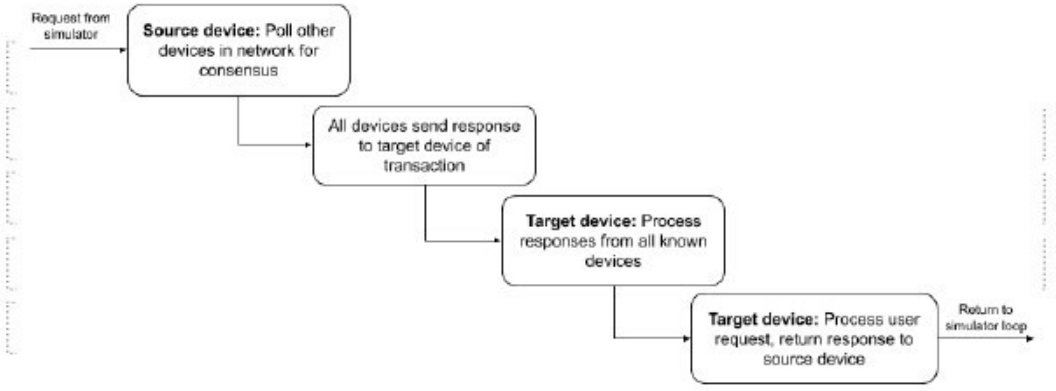

Figure 1: Process Flow in Consensus Model

Upon receiving a request from a simulator instance, the consensus flow requires the source device to make a decision poll to every other device, including the target device, in the network. All devices then send a decision to the target of the request, which evaluates responses and finally processes the user request. The control flow is then returned to the instance thread. Figure 1 is a visual representation of this process.

3.2. Process Flow in ICN Model

Information Centric Networking (ICN) is an approach to Internet infrastructure that focuses on identified information and end-to-end connectivity. This means that content is accessed by name and caching is universal in the network. This means that the one or more devices in an IoT network must act as a name server and cache, making ICN as a heterogeneous network model. ICN also allows easier connectivity to contemporary cloud based models, as the name server within an IoT network can also act as a relay to the cloud server.

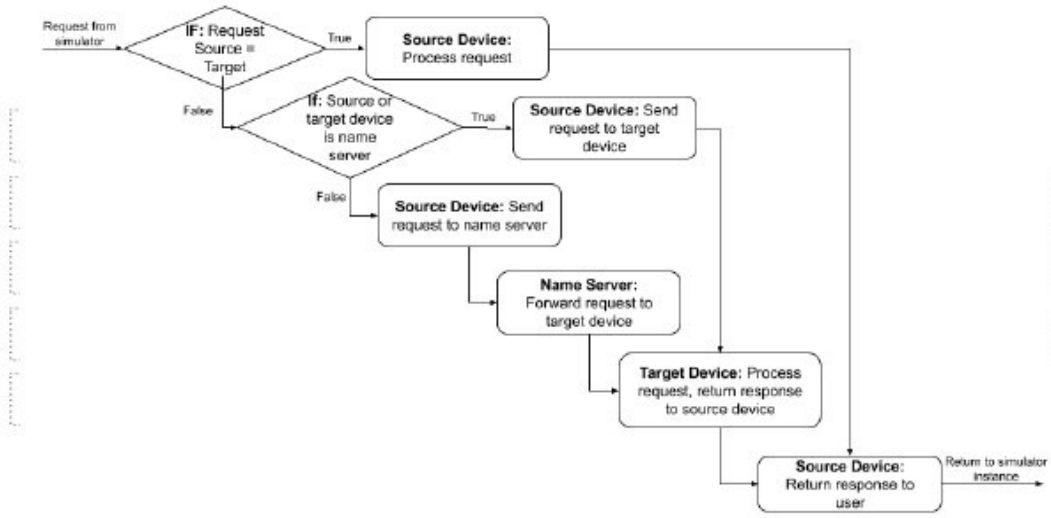

Figure 2: Process Flow in ICN model

Based on how ICN works as depicted in Figure 2, there are three cases that need to be dealt with in the simulator. The first case is when the request source and target device are the same. All that is required is for the source device to process the request, and return a response. The second case is when the name server contacts the intended destination and expects a response. Here, the request source, i.e, the name server, sends the request to the request destination. The destination processes the request and sends back a response. The third case is when the source of the request contacts the name server. Here, the source device (not the same as the destination or the server) sends a request to the name server. The server forwards it to the intended destination. The destination processes the request and sends back a response to the server. The server finally forwards the response to the source device.

3.3. Process Flow in Blockchain based Model

The implementation of blockchain employs Blockchain routers and consists of Full nodes and Lightweight nodes, similar to an Ethereum blockchain. Full nodes have the complete data about other nodes in the chain, and maintain the entire blockchain ledger whereas Lightweight nodes only hold knowledge of full nodes, and only maintain a small section of the ledger. The devices with blockchain are generally autonomous with respect to reads, with the exception of lightweight nodes requesting a full node for data when it does not have the required data in its ledger. On an update in the ledger, the update is broadcast among full nodes, but not lightweight nodes. Hence, lightweight nodes also need to periodically poll a full node to check if its ledger is up to date.

Here, there are 2 main cases to be considered as shown in Figure 3. The first case is when the source device is a Full node and second is a Lightweight node. Each case has a few different scenarios. In the first case where the source device is a full node, the device can move directly to processing the request. In the second case where the source device is a lightweight node, the device must check whether it holds enough information in its partial ledger to be able to make a decision on processing the request. If it does not have this information, it must request a full node for this. After the source device performs a check to ensure that it can process the request, it does so directly if the target device is the same as the source device, or forwards the request to the target device for processing. After the request is processed, the control flow is returned to the simulator instance that running the request.

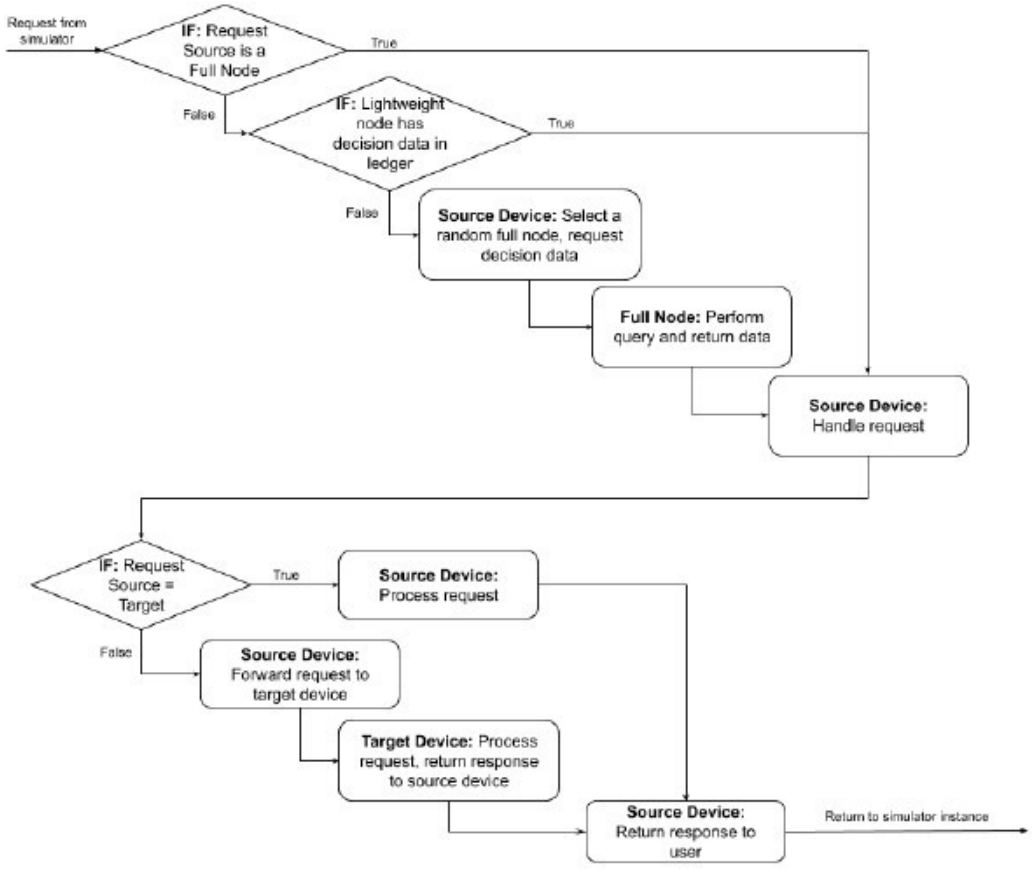

Figure 3: Process Flow in blockchain model

4. Simulation and Performance Analysis

This section evaluates the performance of distributed IoT models under varying inputs. First, the simulation model is elaborated and then performance of three communication model are analyzed.

4.1. Simulation Model

The simulator consists of a number of components and classes that all work together: Devices, Connections, Logger, Simulator Attributes file, and the overall Simulator controller class as shown in Figure 4. Each device in the network is represented by an eponymous Device class. Each object of this class has an id, the total number of messages it has sent and received, the most 15 recent received value. Each device has to receive a message, process the request and send a response. This mechanism is handled through various functions in the class.

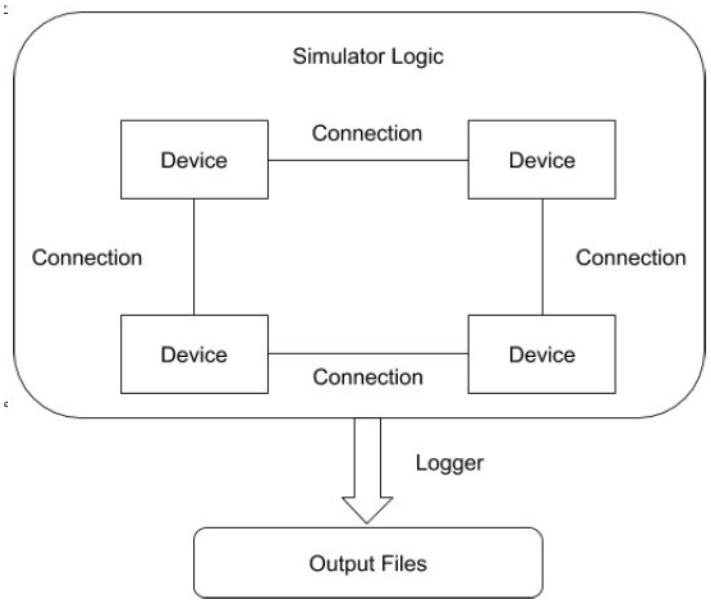

Figure 4: Simulation Model

To represent the connections among devices in the network, a Connection class is used for every device-device pair, that contains mechanisms which allow it to act as a Monitor for synchronization through a producer-consumer process with the paired Devices. It also handles the actual transfer of information among Devices, and represents the network in the simulation.

Synchronization is a key concern that is addressed by Device and Connection class structure. Having a single Device instance per logical device in the IoT system allows to use Java synchronization techniques to ensure that a device isn't working on multiple actions at the same time. The Connection instances are used to represent a virtual mesh network, and synchronize send/receive of data between devices by acting as a Monitor class.

A logging system monitors all events that occur through the course of a simulation run, and writes them to two log files in csv format. The event log contains details of every event that occurred, including every communication between devices and every processing event on devices. The transaction log only stores the details of a transaction that has occurred starting from when the request is generated, to when it has been completed. To set the state of the network, several variables are used. These are stored in a ServerAttributes class. The values here can be changed by the user based on the state of the network that they want to test.

4.2. Performance Analysis under varying device count

Device count refers to the number of devices that are present within the IoT network. Independently changing this variable shows how the communication models perform when in a small or large network, and how increasing the number of devices within the network affects performance.

4.2.1. Average Response Time

From Figure 5, it is clear that the consensus model scales linearly with the number of devices used, which is consistent with the fact that all devices need to be communicated with, and the response of all of them must be evaluated, for each request that is made. Therefore, the consensus model is only suitable for small networks, and preferably where response time does not have a direct impact on user

experience. It is inferred from Figure 6 that the Blockchain model is closer to ideal performance than the ICN model and the trend in their response time does not change much.

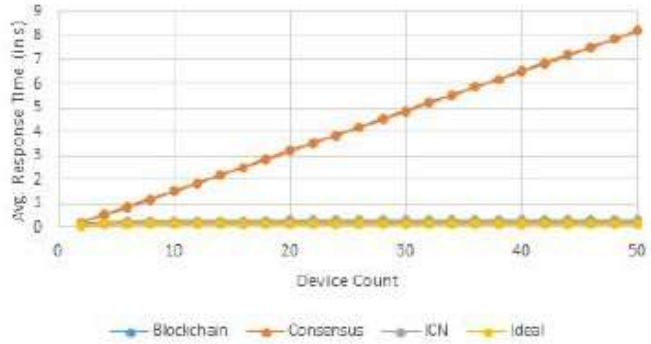

Figure 5: Average Response Time Vs Device Count (For all models)

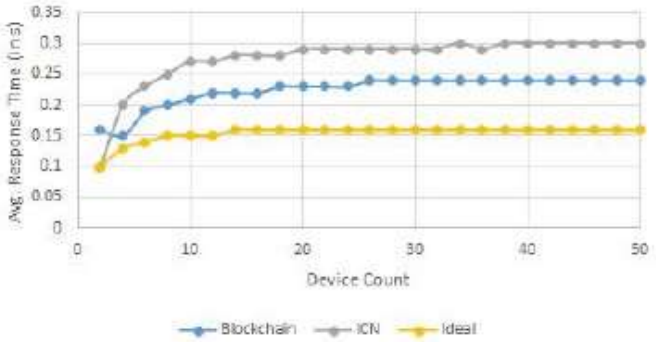

Figure 6: Average Response Time Vs Device Count (Except for Consensus Model)

4.2.2. Throughput

This metric portrays the number of requests that are fully processed by the model per second. It showcases the load handling capacity of evaluated models, and is a significant factor in examining scalability. As shown in Figure 7, throughput response for varying device count is similar to average response time, with blockchain performing better than ICN, and consensus scaling terribly. The blockchain model shows better throughput for larger networks, as it is able to forward requests directly to the target device when a lightweight node that is contacted holds authentication information.

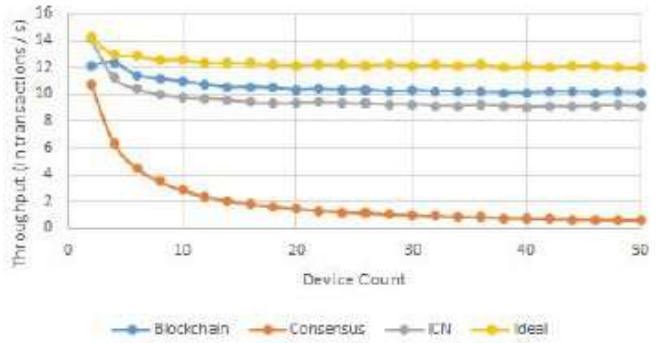

Figure 7: Throughput Vs Device Count

4.2.3. Average Load

This value is a percentage value determining how much load is placed on devices within the network. It establishes the performance and power requirements of devices within the network. It is inferred from Figure 8 that the ICN model performs close to ideal on average, but the central node has a much higher load than the rest, leading to a high max load value. The consensus model has the highest average load of all models, but this load is well distributed among devices, leading to the second lowest max load. The blockchain model shows higher average loads and the highest peak device load, due to the additional cost of performing blockchain operations and the larger responsibilities of the few full nodes in the network. In general, the increase in number of devices results in a subsequent decrease in load, as transactions are more spread out between devices.

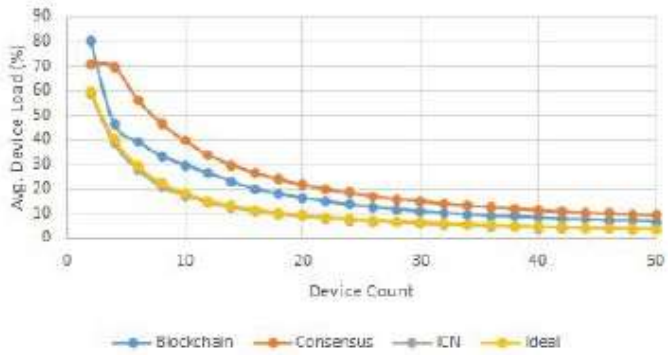

Figure 8: Average Device Load Vs Device Count

Overall, it can be observed that the blockchain model performs well in large networks when each device is capable of handling high loads, and the ICN model being the good performer in a scenario where only one of the devices in the network can sustain high loads.

4.3. Performance analysis under varying Application Weight

Application weight is a variable that accounts for the performance requirements of applications that are used in an IoT environment. This can vary significantly based on how powerful each device is, and how data and performance intensive the application used is. The time taken for an application to finish execution is used as a measure of application weight. An application that takes little time would be considered lightweight, while applications requiring a lot of processing time are heavyweight.

4.3.1. Average Response Time

When examining the models' performance against the cost of performing application specific tasks, the consensus model shows the worst performance overall (as shown in Figure 9) with heavyweight applications by a large margin. In Figure 10, Consensus model is removed to show trends between the other models more clearly, with ICN starting off worse with lightweight applications, but quickly approaches ideal levels of performance for heavy applications, while blockchain performs inversely, showing a larger deviation from ideal for heavier applications due to the system hitting capacity.

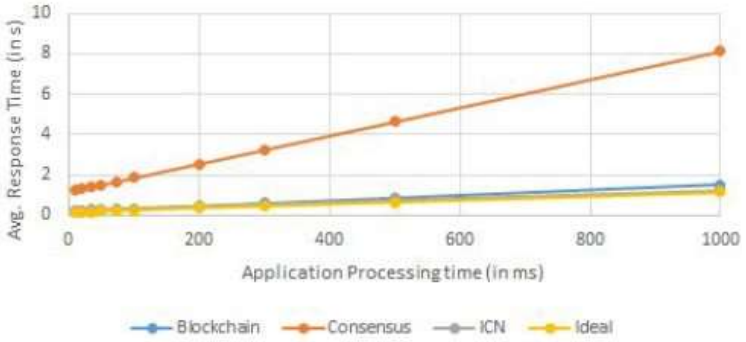

Figure 9: Average Response Time Vs Processing time (for all models)

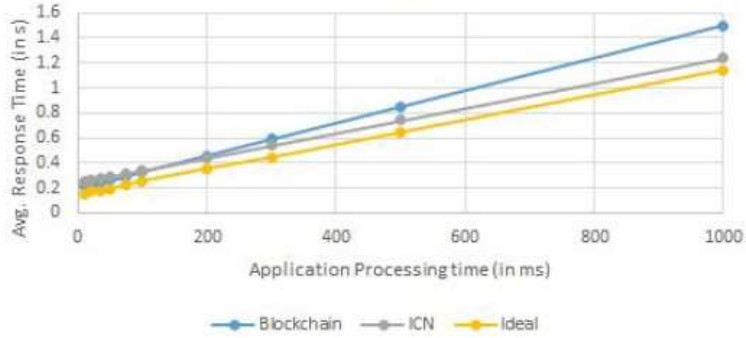

Figure 10: Average Response Time Vs Processing Time (Except for Consensus Model)

4.3.2. Throughput

Throughput against application weight generally follows the trends followed by the models when response time is compared as shown in Figure 4.11. It is noticed that ICN initially performs worse than Blockchain, but converging to ideal levels of performance as application requirements intensify. The consensus model once again underperforms.

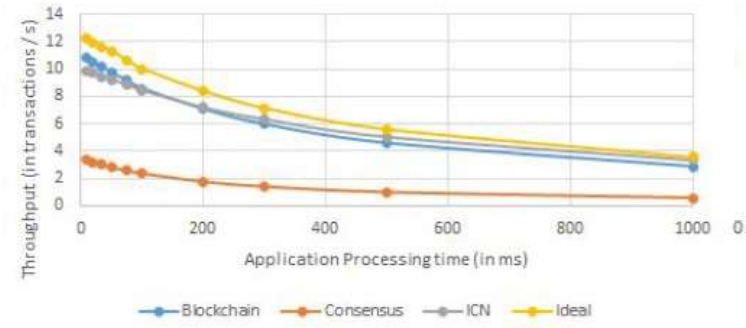

Figure 11: Throughput Vs Processing Time

4.3.3. Average Device Load

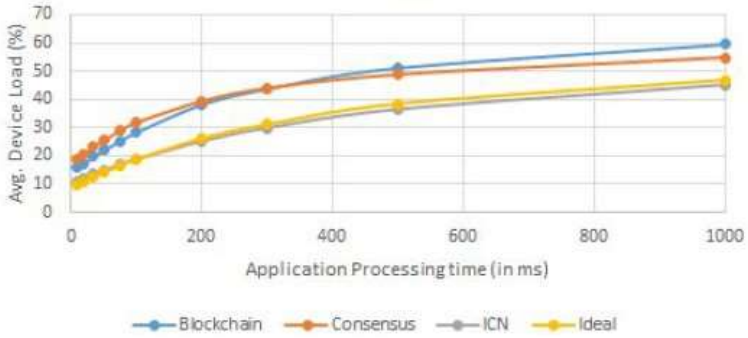

Figure 12: Average Device Load Vs Processing Time

Looking at device loads, the blockchain model shows the worst level of strain put on devices in the network as depicted in Figure 12. It initially performs better on average loads compared to the consensus model, but scaling worse over heavier applications. This is especially seen as one device reaches max load when the application takes ~1s to process, clearly showing the limitations of the blockchain model. The ICN model performs ideally in terms of average load over all devices, but shows the worst performance when lightweight applications are considered. On the whole, ICN performs better for heavyweight applications, whereas the blockchain works better with light application loads.

5. Conclusion

This paper aims to address the growing pains that IoT is facing in terms of scalability, by moving from the centralized, cloud based approach to a distributed environment. Even though, final communication would involve direct communication with the target device, the network model decides how the IoT system would handle the authentication and response to user requests. To address this issue, this paper looks at different distributed IoT models, and compared their performance in a variety of scenarios with an ideal benchmark (P2P that involves no overhead). From the simulation results, it is inferred that the ICN model is the better model for small networks than other models, but it places a higher load on the central device that acts as a name server, and generally has a slightly worse response time and throughput compared to other models. The Blockchain has the potential to offer the best security in terms of implementation. However, the performance and security comes at the cost of higher processing loads on all devices, especially on the full nodes in the network. As such, it is unsuitable for use with low power IoT devices. The Consensus model is the most balanced of the three in terms of device loads, and is suitable for when only a few low power devices are in use. In most other cases, the consensus model fails to perform well, including in larger networks, for higher user loads, higher latency networks, or more intensive applications.

6. References

[1] W.Z. Khan, M.H. Rehman, H.M. Zangoti, M.K. Afzal, N. Armi, K. Salah, ”Industrial internet of things: Recent advances, enabling technologies and open challenges”, Computers & Electrical Engineering, 81, (2020), ISSN 0045-7906.

[2] G. Fersi, “Middleware for Internet of Things: A Study”, International Conference on Distributed Computing in Sensor Systems (2015), doi: 10.1109/DCOSS.2015.43.

[3] W. Mahmood and A. Wahab, “Survey of Consensus Protocols”, ArXiv abs/1810.03357 (2018).

[4] B. Nour, K. Sharif, F. Li and H. Moungla, "A Distributed ICN-Based IoT Network Architecture: An Ambient Assisted Living Application Case Study," IEEE Global Communications Conference, Singapore, 2017, pp. 1-6.

[5] N. Chaudhry and M. M. Yousaf, “Consensus Algorithms in Blockchain: Comparative Analysis, Challenges and Opportunities,” 12th International Conference on Open Source Systems and Technologies (ICOSST), Pakistan, 2018, pp. 54-63, doi: 10.1109/ICOSST.2018.8632190.

[6] Mars, D., Mettali Gammar, S., Lahmadi, A. *et al.* Using Information Centric Networking in Internet of Things: A Survey. *Wireless Pers Commun* 105, (2019), 87–103.
[7] X.M. Zhang and N. Zhang, "An open secure and flexible platform based on internet of things and cloud computing for ambient aiding living and telemedicine", International Conference on Computer and Management, China, 2011, pp. 1-4, doi: 10.1109/CAMAN.2011.5778905.
[8] S. Li, Y. Zhang, D. Raychaudhuri, R. Ravindran, Q. Zheng, L. Dong, et al., "IoT Middleware Architecture over Information-Centric Network", Globecom Workshops (2015) pp. 1-7.
[9] R. Ravindran, T. Biswas, X. Zhang, A. Chakraborti and G. Wang, "Information-centric networking based homenet," 2013 IFIP/IEEE International Symposium on Integrated Network Management (IM 2013) , Ghent, 2013, pp. 1102-1108.
[10] W. Shang et al., "Named Data Networking of Things (Invited Paper)," 2016 IEEE First International Conference on Internet-of-Things Design and Implementation (IoTDI) , Berlin, 2016, pp. 117-128
[11] M. A. Hail and S. Fischer, "IoT for AAL: An Architecture via Information-Centric Networking," 2015 IEEE Globecom Workshops (GC Wkshps) , San Diego, CA, 2015, pp. 1-6.
[12] A. Gervais, G. O. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, and S. Capkun, "On the Security and Performance of Proof of Work Blockchains," Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, New York, 2016, pp. 3–16. doi:https://doi.org/10.1145/2976749.2978341
[13] . A. D. G. Dimakis, A. D. Sarwate, and M. J. Wainwright, "Geographic gossip: efficient averaging for sensor networks," IEEE Transactions on Signal Processing , 56,3, (2008), 1205–1216.
[14] I. Abraham, D. Malkhi et al., "The blockchain consensus layer and bft", Bulletin of EATCS , 3, 123, 2017.
[15] Crain, Tyler, Vincent Gramoli, M. Larrea and M. Raynal. "(Leader/Randomization/Signature)-free Byzantine Consensus for Consortium Blockchains." ArXiv abs/1702.03068 (2017).
[16] H. Wang, Y. Cen, and X. Li, "Blockchain Router," Proceedings of the 6th International Conference on Informatics, Environment, Energy and Applications, New York, 2017, pp: 94–97.
[17] D. Ding, "InterChain : A Framework to Support Blockchain Interoperability." 2018.
[18] W. Li, A. Sforzin, S. Fedorov, and G. O. Karame, "Towards Scalable and Private Industrial Blockchains," Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts – BCC 17, New York, 2017, pp: 9-14.
[19] G. G. Dagher, C. L. Adhikari, and T. Enderson, "Towards Secure Interoperability between Heterogeneous Blockchains using Smart Contracts," Future Technologies Conference (FTC), Canada, 2017, pp:73-81.
[20] A. Culwick and D. Metcalf, "The Blocknet Design Specification.", https://www.blocknet.co/wp-content/uploads/whitepaper/BlocknetWhitepaper.pdf.