

Simulators, Emulators, and Test-beds for Internet of Things: A Comparison

N D Patel

Centre of Excellence
in Cyber Security (IDRBT),
Hyderabad 500057, India &
University of Hyderabad, India
Email: ndpatel@idrbt.ac.in

B M Mehtre

Centre of Excellence
in Cyber Security (IDRBT),
Castle Hills, Masab Tank,
Hyderabad 500057, India
Email: bmmehre@idrbt.ac.in

Rajeev Wankar

School of Computer and
Information Sciences (SCIS),
University of Hyderabad
Gachibowli, Hyderabad 500046, India
Email: wankarcs@uohyd.ernet.in

Abstract—In the last few years, the world has seen extensive growth in the field of IoT in both research & deployment and adoption of the Internet of Things (IoT). The number of IoT devices is increasing multi-fold each year. So, the state of IoT is under rigorous research & deployment by several researchers across the globe to develop new algorithms, techniques, and protocols to secure IoT networks further. In this paper, we compare sixteen simulators (*IOTSim, IoTIFY, Bevywise-IoT, etc*), four emulators (*Cooja, NetSim, MAMMoH, NCTUns 6.0*), and six test-beds (*MBTAAAS, FIT IoT-LAB, etc*) for IoT on the basis of Scope, Type, Programming language, IoT layers, Scale of Operation, IoT Standards, API Integration, Cyber Resilience, Service Domain, and Security Measures. Its comparison helps for modeling techniques, proper simulation, and testing for algorithm validation/proof-of-concept implementation in IoT. This comparison will help to support researchers and developers for choosing suitable tools for their application. We have also discussed Vertical & Horizontal Analysis of existing Simulators, Emulators, and Test-beds. This will help in testing the research ideas before moving towards live implementation.

Keywords: IoT Emulators, IoT Simulators, IoT Test-beds

I. INTRODUCTION

In IoT, advanced computing and electronic technology will establish a smart atmosphere by utilizing smart-sensors/actuators/things/devices/objects to connect and communicate the smart ecosystem with social and user contexts. When we are talking about the IoT, it is a paradigm shift of upcoming technology for IoT applications. It is just talking about how can we put these IoT devices onto almost anything or anywhere and then what can we do with it. There are many IoT sensors like accelerometer, pedometer, temperature sensor, GPA sensor, etc [1]. They communicate in a lot of ways to connect computers at a Local Area Network (LAN) using different protocols [2]. Nowadays, many smart devices use wireless Ethernet and TCP/IP connection in the IoT Ecosystem.

Kumar et al., [3] gave an overview of IoT architecture, where they have reviewed communication protocols, application issues, and the latest growth of RFID. ND Patel et al., [4] proposed Things-to-Cloud (T2C) a nine layered protocol-based architecture.

In the research, we need a strategy that fulfills following distinctive criteria like repeatability, experimental control,

realism, configurability, automation, behavior modification, remote administration, isolation (from arranged systems), scalability, and integration with wired test-beds & systems. Three techniques are presently accessible for IoT environment: Simulators, Emulators, and Test-beds. Simulator is a "device that enables the operator to reproduce or represent under test conditions phenomena likely to occur in actual performance [5]". Emulator is a "hardware or software that permits programs written for one computer to be run on another computer [5]". Test-bed, which is "based on the immersion of system components under test in a virtual environment, in order to enable testing of IoT applications [6]".

To design algorithms, testing application protocols, verification of the environment, and real-time implementation, researchers required pre-knowledge about simulators, emulators, and test-beds to choose according to their needs. There are many benefits of simulators, emulators, and test-beds like configurability, repeatability, manageability, scalability, integration, isolation, and controlling. They will allow a level of control over the real-world, use of real-time devices, and communication with the OS. They will provide a level of scaling and the ability to run on real-time based IoT applications.

This paper presents a survey of existing IoT research tools (simulators, emulators, and test-beds) that attracted the attention of researchers to provide a proof-of-concept and specifications model, sample, or release of the life-cycle. The rest of the paper is organized as follows: in section II, we compare the IoT Simulators. In section III, we compare the IoT Emulators. In section IV, we compare the IoT open test-beds. In Section V, we analyze and discuss on Vertical & Horizontal Analysis. Section VI gives the conclusions and the direction for future scope.

II. IOT SIMULATORS

Simulation means analysis and study of various tools and techniques. A simulator is a particular specific system or computer programming model that simulates real-time events and manages to recognize the outcomes or consequences of various hypotheses or assumptions, without exhibiting the experimenter to any uncertainty danger or risk. The Cambridge-dictionary defines a simulator as: "a bit of equipment that

is designed to represent real conditions". Many simulators have been designed with Graphical User Interface (GUI) and primarily based engine, so they allow researchers to get started quickly with the simulation process. It also helps in assessing IoT simulators and contributions to the current pool of IoT devices. The IoT simulator helps the users to simulate hundreds of linked gadgets that are defined by the users without creating any problems. There are so many advantages of simulators like control, configurability, manageability, integration, repeatability, and scalable. IoT simulators allow various devices to interact or communicate with each other continuously. IoT device testing and examination is a method of recognizing or classifying failures, wherever a breakdown consists of any difference between original and supposed results. IoT simulator is helpful in making the actual IoT environment to simulate thousands of connected devices effortlessly as compare to complicated connection of physical IoT devices.

A comparison of top sixteen simulators is given in table number I. This table comprises the following parameters such as scope, type, programming language, IoT architecture layers, scale of operation, build-in IoT standards, API integration, cyber resilience simulation, service domain, and security measures.

1) **DPWSim [7]**

DPWSim is a device profile for web services that highlighting secure messaging, effective identification, service classification. It is a simulation toolkit for IoT devices and applications. DPWS determines a model in which devices test two types of services: Hosted Services and Hosting Services. It specifies a secure web services messaging, detection, interpretation, and resource-constrained devices.

2) **iFogSim [8]**

iFogSim is an Edge and Fog Computing Simulator on the top of cloud simulator to capable of simulating IoT, fog data centers, edge devices, network connections, cloud analysis, and perform metrics assessment approaching them. Among these characteristics, it offers research examination and contrast of system management techniques/tactics constructed on QoS standards (e.g., network congestion, latency, bandwidth, task scheduling, etc.).

3) **IOTSim [9]**

IOTSim, it is a cloud-based IoT simulator and developed on top of the *CloudSim*. It is produced to assist the checking out of *Big-Data* processing and address with the *MapReduce* method. By genetically helping significant information policies, it simplifies the recognition & analysis of results and execution of IoT-based purposes through researchers and profitable business industries.

4) **CupCarbon [10]**

It is a multi-agent, 2D/3D Visualization, discrete Wireless Sensor Network (WSN) event debug simulator. It validates the distributed algorithm and automatization of some

repeated tasks for industrial purposes. It can help trainers in exhibiting the essential standards and how smart sensor networks work. It can aid the analyst in trying out their wireless topology, communication, protocols, routing, etc.

5) **SimIoT [11]**

SimIoT is a toolkit to operate function trial and error on static/dynamic or real-time multi-user acquiescence within an IoT scenario. The toolkit is essentially established on the cross hybrid implementation of the *SimIC*. It is a method that in modelers to configure and construct a heterogeneity of clouds in terms of data-center hosts. It is software insurance policies, at which point the preferred quantity of customers ought to send a person or multiple requests for estimation endowment, software program sources, and continuation of virtual machine.

6) **OMNeT++ [12]**

OMNeT++ is a non-commercial simulation and discrete-event simulator model. It is fundamentally based on a C++ simulation framework and library, notably for building network simulators. It is based on the Eclipse platform. OMNeT++ can be used for free in educational institutions. OMNeT++ assembled into larger components and techniques using a High-Level Language (HLL). It is an "object-oriented" and "discrete event" network simulation model, which is a generic architecture used in different apps.

7) **NS-Series [13]**

Network Simulator (NS) is a title for a sequence of distinct various event community network simulators for networking research, the mainly "NS-1, NS-2, NS-3, and NS-4" are distinct-event computer-based network simulators, they support a bunch of protocols like FTP, TCP, UDP, DSR, and DSR. NS-Series simulators are free software, authorized under the "GNU GPLv2" authorize, and is publicly available for everyone.

Kumar et al., [23] presented an efficient resource allocation method and quality of service (QoS) enhancement to obtain information on the proficiency method in the fog using NS-2 simulator.

8) **QualNet [14]**

The QualNet simulator with the aid of flexible networks. This simulator can maintain large accuracy simulations, including various network elements. IoT based simulation can be performed by takes the other smart sensor network development for QualNet, which provides the guidelines for the "IEEE 802.15.4 community series". It is a commercial, profitable version of Glomosim with a GUI feature.

9) **Atomiton IoT Simulator [15]**

The Atomiton IoT Simulator constructed atop Atomiton Stack (a specific accomplishing operating environment situation for the IoT), is checking out the model able to simulate smart digital sensors, gadgets, and actuators with unique behaviors for an appropriate node.

10) **SWE Simulator [16]**

This simulator helps you designing and checking out

Simulators	Scope	Type	Programming Language	IoT Architecture Layers	Scale of Operation	Built-in IoT Standards	API Integration	Cyber Resilience Simulation	Service Domain	Security Measures
DPWSim [7]	IoT	Open Source	Java	Application	Small scale	Secure Web Services Messaging	SOAP	No	Generic	Medium
iFogSim [8]	Fog	Discreteevent	Java	Perceptual Network Application	Not known	No	SOAP	No	Generic	Medium
IOTSim [9]	Data analysis	MapReduce model	Java	Application	Large scale	No	REST	No	Generic	Medium
CupCarbon [10]	Network	Agent-based discreteevent	Java Custom scripting	Perceptual Network	Small scale	802.15.4 LoRaWAN	UDX	No	Smart City	High
SimIoT [11]	Data analysis	Discreteevent	Java	Application	Small scale	No	REST	No	Generic	High
OMNeT++ [12]	Network	Discreteevent	C++	Perceptual Network	Large scale	Manual extension	SOAP	Custom extensions	Generic	Medium
NS-Series [13]	Network	Discreteevent	C++	Perceptual Network	Large scale	802.15.4 LoRaWAN	REST	No	Generic	High
QualNet [14]	Network	Discreteevent	C++	Perceptual Network	Large scale	802.15.4 (Zigbee only)	REST	Yes	Generic	Medium
AtomitonIoT [15]	IoT IIoT	Edge Anatany	Go Java	Communication Network	Large Scale	Socialize	REST	No	MQIdentity	High
SWE-IoT [16]	WSN	Sensor Observation Service	C C++	Communication Network	Small Scale	Collision detection	SOAP	No	Human Interface	High
TOSSIM [17]	TinyOS	Sensor Observation Service	C Python	Communication Network	Small Scale	Injecting Packets	REST	Yes	Generic	High
MobIoTSim [18]	IoT Networks	Research Based	C++ C Sharp	Application Network	Large scale	Devices Profile for Web Services (DPWS)	REST	No	Generic	Medium
EdgeCloudSim [19]	Edge WLAN	Realistic	Matlab	Network	Large scale	Mist Computing	SOAP	No	Edge Orchestrator	High
IoTIFY [20]	Hardware Connection	Mobile App	Python Java	Application Network	Large scale	Real Time	REST	Yes	Smart City	High
Bevywise-IoT [21]	IoT Device	Broker	Python Java	Network	Large scale	real Time	REST	No	Smart City	Medium
Ansys-IoT [22]	IoT Industry	Autonomous	Python Java	Network	Large scale	real Time	REST	Yes	Industry	High

Table I: Comparison of Selected IoT Simulators

large sensors networks that can be carried out through the potential of simulation tools. They facilitate quicker checks barring risks and also minimize the cost for implementation. Its simulation and testing performance for the IoT devices, as well as designing for a broad range of sensor networks, based on sensor observation services (SoS).

11) **TOSSIM [17]**

TOSSIM is a smart WSN/IoT simulator, and it is built along with the different causes to simulate *TinyOS* smart devices. Since *TinyOS* is an action-based operating system, it is beside problems interpreted into a simulator appliance with distinct events; for that reason, facilitate it and performing it more productive and efficient.

12) **MobIoTSim [18]**

MobIoTSim is a mobile-based simulator for IoT devices. It's developed in the android platform, designed to assist researchers in determining IoT devices having smart sensors and demonstrate IoT applications utilizing more than one device.

13) **EdgeCloudSim [19]**

EdgeCloudSim addresses the particular necessities of the Edge Computing query and redesigns the required usefulness as far as calculation and systems administration capacities. To demonstrate the capacities of EdgeCloudSim, a test arrangement dependent on exceptional viewpoint designs is reenacted, and the impact of the computational and systems administration machine parameters on the results are depicted.

14) **IoTIFY [20]**

IoTIFY is a software extension surroundings for IoT besides embedded hardware territories. By resorting to the device virtualization and intelligent device simulator, it affords a digital lab for building analytical modeling, embedded prototypes, load testing solution, and a community simulation for analytical scaling, virtual devices in the fog environment, and records generation.

15) **Bevywise IoT Simulator [21]**

Bevywise IoT Simulator supports your experiment with on-premise MQTT (virtual clients) application in fog computing environment for users and functional testing on the cloud. We can simulate thousands of the com-

modity server. Bevywise simulator provides an end to end product and solution for IoT & Industrial-IoT. It can be used to demo, develop, and test in a real-time IoT environment.

16) **ANSYS-IoT Simulator [22]**

ANSYS-IoT simulator can help to the various challenges of the IoT devices. Discover how to validate and even improve the reliability, power consumption, longevity, and overall integrity of smart sensors in IoT devices.

III. IOT EMULATORS

An emulator is a software or hardware program that enables one system to behave like other computing devices for testing. Emulator usually allows the host device to run the software program or use peripheral devices designed for the guest gadget. Emulators are a middle ground between real simulation and wireless test-beds. It consolidates the configurability, manageability, repeatability, and the authenticity of test-beds. IoT Emulators regularly empower the host framework (system) to run programming or utilize peripheral gadgets intended for the visitor framework (system). A comparison of top four emulators is given in table number II. This table comprises the following parameters such as scope, type, programming language, IoT architecture layers, scale of operation, build-in IoT standards, API integration, cyber resilience simulation, service domain, and security measures.

1) **Cooja [24]**

The cooja emulation/simulation is a network platform produced explicitly for the "Contiki OS" & "WSN". It is a java-based simulator and capable of simulating the IoT sensors (Z1 sensors = sense/send/blink), RPL in WSNs (example = UDP-RPL/broadcast), preparation set, system controlled, analyzed, and operating system. Cooja emulate the pleasurable of the corresponding firmware update. It is additionally uploaded to the fog nodes and alternatively of simulate IoT devices. It provides to researchers to take a look at their code and simple debugging exercise.

2) **MAMMotH [25]**

MAMMotH is a small/large scale based real-time IoT emulator, being adequate to emulate thousands of devices per virtual machine. The mobile gadgets also related to the aid of GPRS to a base station, which in turn join to in the back-end as a large scale IoT Emulator. It is a massive scale emulation platform for IoT with cost-effective approach to test small and large scale networks infrastructure and support desired IoT applications.

3) **NetSim [26]**

NetSim is an advanced IoT network simulation software for protocol modeling/simulation, network devices, and security applications. It allows you to investigate computer networks with unmatched intensity, power, adaptability, and flexibility. NetSim (discrete event simulator) covered an extensive range of wired, mobile networks, wireless, IoT devices, and sensor networks. It is used by

50+ types of research globally for IoT, IIoT, WSN, LTE, MANET, etc.

4) **NCTUns 6.0 [27]**

NCTUns (National Chiao Tung University Network Simulator) is an open source discrete event based network simulator cum emulator. It runs on Linux platform and gives GUI network environment for testing & development.

IV. IOT TEST-BEDS

Test-bed, gives researchers a wide scope of IoT environments where they can create, develop, debug, troubleshoot, and analyze their models. The advantages of test-beds are, it allows to have dimension of authority over the present reality, it uses real-time gadgets, real layer interaction with the OS, it also gives a dimension of scaling, remote administration, and ability to run real-time applications. There are few inconveniences present in the test-beds like: repeatability, scaling, mobility of hubs, and dependant on the area.

IoT test-beds enables different gadgets to associate or speak with another one while performing testing and an examination of those gadgets. These test-beds are use to create IoT applications without using physical devices for smart homes, smart cities, and other applications. Test-beds control system will be connected to your test, and the potential system is programmed to monitor all IoT devices and gadgets. We can easily analyse communication data during the test, and can report any exceptions happened during the test.

A comparison of top six test-beds is given in table number III. This table comprises the following parameters such as scope, type, programming language, IoT architecture layers, scale of operation, build-in IoT standards, API integration, cyber resilience simulation, service domain, and Support for Visualizations.

1) **MBTAAS [28]**

MBTAAS is the model-based testing as a service. It provides the typically take a look at the IoT and learning information platforms. The access lodges to an aggregate of framework based testing strategies and service oriented solutions. We can test our cloud and premises.

2) **WHYNET [29]**

Wireless HYbrid NETwork (WHYNET) testbed, presently being developed for heterogeneous, large scale based, realistic WSN protocols, scalable evaluation of next-generation and applications based test-beds. WHYNET enables consistent integration of physical and emulation components in a single model. It is a hybrid test-beds for small-scale, large-scale, miscellaneous, and adaptive IoT networks.

3) **FIT IoT-LAB [30]**

FIT IoT-LAB (Federated IoT test-bed for large scale experimentation) is an objective of experimental test-bed for trying out a large scale WSN/IoT/Embaded_devices/Gadgets. It can be broadcasting various objects and constructed on a very massive scale framework. They have been deployed

Emulators	Scope	Type	Programming Language	IoT Architecture Layers	Scale of Operation	Built-in IoT Standards	API Integration	Cyber Resilience Simulation	Service Domain	Security Measures
Cooja [24]	Network	Discrete-event	C/Java	Perceptual Network	Small scale	Protocols supported by Contiki OS	REST	Custom infrastructure	Enables real world	High
MAMMotH [25]	IoT Device	M2M	Python Java	Application Network	Large scale	Cost-efficient	REST	No	Generic	Medium
NetSim [26]	IoT Networks	Research Based	C Code	Perceptual Network	Campus-Wide Use	802.15.4 LTE MANETs	SOAP	Yes	Military Utilities	High
NCTUns 6.0 [26]	Sensor Networks	Discrete-event	C++	Network Data-Link	Large Scale	802.11p WiMAX MANETs Optical Network	SOAP	Yes	Open Source	High

Table II: Comparison of Selected IoT Emulators

Test-beds	Scope	Type	Programming Language	IoT Architecture Layers	scale of operation	Built-in IoT Standards	API Integration	Cyber Resilience Simulation	Service Domain	Support for Visualizations
MBTAAS [28]	IoT Platform	Service Oriented	OCL	All	Large scale	Model Based	REST	No	Smart City	IoT Dashboard
WHYNET [29]	Wireless Network	Network Protocol	Java	Network	Large scale	Application Based	SOAP	No	Energy Efficient	Web Portal
FIT IoT-LAB [30]	IoT Network	IoT Spectrum	nesC Java	Perceptual Network	Small Scale	802.15.4 LoRaWAN	REST	No	Heterogeneous Platform	FIT Cloud
FIESTA-IoT [31]	Energy	Sensor Observation Service	C, Java Python	Communication Network	Large Scale	Energy Consumption	REST	Yes	Ambient Environment	Meta-Cloud
Smart Santander [32]	IoT Mobile sensing	Map data Map Dataimage	Java JavaScript	Application Network	Large Scale	IEEE 802.15.4 GPRS RFID tags	REST	Yes	Smart City	Management Console
JOSE [33]	IoT WSN SDN	Smart ICT Service Platform	C, Java JavaScript	Virtualized Network Services	Large Scale	Sensor Networks	SOAP	Yes	Real Time	Distributed Cloud

Table III: Comparison of Selected IoT test-beds

around more than 3000 low-power WSN/IoT nodes and more than 200 mobile robots available for the testing environment. It is the replacement of *SENSLAB Test-bed* and the section of the Future-Internet of Things (FIT) platform.

4) **FIESTA-IoT [31]**

FIESTA-IoT is a federated interoperable semantic IoT test-beds & applications for IoT, the FIESTA-IoT platform federates 10 IoT test-beds sponsoring get access to a wealth of various IoT data sets. They offer live web-services over the IoT data sets. FIESTA will open new horizons boundaries in IoT experimentation at a global scale, primarily based on the interconnection and the ability to exchange various IoT test-beds.

5) **Smart Santander [32]**

The smart santander is monitoring the mobile environment, traffic intensity, guide to parking lots, and etc. It considers more than 3000 devices, 200 GPRS, 3000 RFID tag/QR code deployed at static locations as well as on_board of Mobile Specialty Vehicles (MSV).

6) **JOSE [33]**

JOSE offerings as virtualized network services for IoT applications. It is an open test-bed for the outdoor environment with heterogeneity, and JOSE service platform accommodates multiple large-scale subject trials that consist of storage sources, sensor networks, and computation resources.

V. ANALYSIS & DISCUSSION

We have presented stages of the product/software development process in terms of simulators, emulators, and test-beds for pre-testing. A study on "Vertical Analysis" and "Horizontal Analysis" is presented in this section. The first choice that, whether to choose a simulator, emulator, or/and test-bed is based on IoT environment, whether they are acceptable for all phases of testing in different scenarios or not.

A simulator is the right option at the early phase of design, development, and test IoT applications & devices without setup actual IoT boards. Simulators give a higher level of abstraction and test futuristic IoT devices. For example, the design and construction of the Routing Protocol for Low-Power and Lossy Networks (RPL) and topology controller decisions can be performed based on simulation. It requires the high expenditure and expense to implement with hundreds of nodes in the real environment, and also the time, effort, and resources. So simulators are significant tools to construct and test IoT environment and networks in a virtual setup.

Emulators are created 'entire network (real + virtual)' with mapping of real devices to equivalent simulated devices. Mainly they execute the parallel code on the real-nodes, so they reduce real-implementation exercise. The results or outcomes received from emulators are more reliable than simulators. Most of the emulators are based on component-based architecture to allow more extraordinary scalability in IoT environments.

- [9] X. Zeng, S. K. Garg, P. Strazdins, P. P. Jayaraman, D. Georgakopoulos and R. Ranjan, 'Iotsim: A simulator for analysing iot applications', *Journal of Systems Architecture*, vol. 72, pp. 93–107, 2017.
- [10] K. Mehdi, M. Lounis, A. Bounceur and T. Kechadi, 'Cupcarbon: A multi-agent and discrete event wireless sensor network design and simulation tool', in *7th International ICST Conference on Simulation Tools and Techniques, Lisbon, Portugal, 2014*, 2014, pp. 126–131.
- [11] W. Osborn, '2014 28th international conference on advanced information networking and applications workshops (waina)',
- [12] A. Varga and R. Hornig, 'An overview of the omnet++ simulation environment', in *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*, 2008, p. 60.
- [13] T. R. Henderson, M. Lacage, G. F. Riley, C. Dowell and J. Kopena, 'Network simulations with the ns-3 simulator', *SIGCOMM demonstration*, p. 527, 2008.
- [14] S. Siraj, A. Gupta and R. Badgujar, 'Network simulation tools survey', *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 1, no. 4, pp. 199–206, 2012.
- [15] <https://atomiton.atlassian.net/wiki/spaces/TQLDocs/pages/15728645/TQL+Approach+to+IoT+Simulation>.
- [16] P. Giménez, B. Molina, C. E. Palau and M. Esteve, 'Swe simulation and testing for the iot', in *2013 IEEE International Conference on Systems, Man, and Cybernetics*, IEEE, 2013, pp. 356–361.
- [17] P. Levis and N. Lee, 'Tossim: A simulator for tinyos networks', *UC Berkeley, September*, vol. 24, 2003.
- [18] T. Pflanzner, A. Kertész, B. Spinnewyn and S. Latré, 'Mobiotsim: Towards a mobile iot device simulator', in *2016 IEEE 4th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW)*, IEEE, 2016, pp. 21–27.
- [19] C. Sonmez, A. Ozgovde and C. Ersoy, 'Edgecloudsim: An environment for performance evaluation of edge computing systems', *Transactions on Emerging Telecommunications Technologies*, e3493, 2018.
- [20] Z. Wang and Y. Shen, 'Job-aware scheduling for big data processing', in *Cloud Computing and Big Data (CCBD), 2015 International Conference on*, IEEE, 2015, pp. 177–180.
- [21] P. Pięta, S. Deniziak, R. Belka, M. Płaza and M. Płaza, 'Multi-domain model for simulating smart iot-based theme parks', in *Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2018*, International Society for Optics and Photonics, vol. 10808, 2018, 108082T.
- [22] <https://www.ansys.com/en-in/campaigns/internet-of-things>.
- [23] T. S. Kumar, 'Efficient resource allocation and qos enhancements of iot with fog network', *Journal of ISMAC*, vol. 1, no. 02, pp. 101–110, 2019.
- [24] F. Osterlind, A. Dunkels, J. Eriksson, N. Finne and T. Voigt, 'Cross-level sensor network simulation with cooja', in *Local computer networks, proceedings 2006 31st IEEE conference on*, IEEE, 2006, pp. 641–648.
- [25] V. Looga, Z. Ou, Y. Deng and A. Yla-Jaaski, 'Mammoth: A massive-scale emulation platform for internet of things', in *Cloud Computing and Intelligent Systems, 2012 IEEE 2nd International Conference on*, IEEE, pp. 1235–1239.
- [26] P. Nayak, 'Comparison of routing protocols in wsn using netsim simulator: Leach vs leach-c', *International Journal of Computer Applications*, 2014.
- [27] S.-Y. Wang, C.-L. Chou, C.-C. Lin and C. Huang, 'The gui user manual for the nctuns 6.0 network simulator and emulator', *National Chiao Tung University*, 2010.
- [28] A. Ahmad, F. Bouquet, E. Fournieret, F. Le Gall and B. Legeard, 'Model-based testing as a service for iot platforms', in *International Symposium on Leveraging Applications of Formal Methods*, Springer, 2016, pp. 727–742.
- [29] J. Zhou, Z. Ji, M. Varshney, Z. Xu, Y. Yang, M. Marina and R. Bagrodia, 'Whynet: A hybrid testbed for large-scale, heterogeneous and adaptive wireless networks', in *Proceedings of the 1st international workshop on Wireless network testbeds, experimental evaluation & characterization*, ACM, 2006, pp. 111–112.
- [30] C. Adjih, E. Baccelli, E. Fleury, G. Harter, N. Mitton, T. Noel, R. Pissard-Gibollet, F. Saint-Marcel, G. Schreiner, J. Vandaele *et al.*, 'Fit iot-lab: A large scale open experimental iot testbed—a valuable tool for iot deployment in smart factories', *IEEE ComSoc Multimedia Technical Committee E-Letter*, 2015.
- [31] A. Gyrard and M. Serrano, 'Fiesta-iot: Federated interoperable semantic internet of things (iot) testbeds and applications', in *ICT*, 2015.
- [32] L. Sanchez, L. Muñoz, J. A. Galache, P. Sotres, J. R. Santana, V. Gutierrez, R. Ramdhany, A. Gluhak, S. Krco, E. Theodoridis *et al.*, 'Smartsantander: Iot experimentation over a smart city testbed', *Computer Networks*, vol. 61, pp. 217–238, 2014.
- [33] M. Chernyshev, Z. Baig, O. Bello and S. Zeadally, 'Internet of things (iot): Research, simulators, and testbeds', *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 1637–1647, 2018.