2016; **00**:1–22

iFogSim: A Toolkit for Modeling and Simulation of Resource Management Techniques in Internet of Things, Edge and Fog Computing Environments

Harshit Gupta * †, Amir Vahid Dastjerdi *, Soumya K. Ghosh†, and Rajkumar Buyya *

SUMMARY

Internet of Things (IoT) aims to bring every object (e.g. smart cameras, wearable, environmental sensors, home appliances, and vehicles) online, hence generating massive amounts of data that can overwhelm storage systems and data analytics applications. Cloud computing offers services at the infrastructure level that can scale to IoT storage and processing requirements. However, there are applications such as health monitoring and emergency response that require low latency, and delay caused by transferring data to the cloud and then back to the application can seriously impact their performances. To overcome this limitation, Fog computing paradigm has been proposed, where cloud services are extended to the edge of the network to decrease the latency and network congestion. To realize the full potential of Fog and IoT paradigms for real-time analytics, several challenges need to be addressed. The first and most critical problem is designing resource management techniques that determine which modules of analytics applications are pushed to each edge device to minimize the latency and maximize the throughput. To this end, we need a evaluation platform that enables the quantification of performance of resource management policies on an IoT or Fog computing infrastructure in a repeatable manner. In this paper we propose a simulator, called iFogSim, to model IoT and Fog environments and measure the impact of resource management techniques in terms of latency, network congestion, energy consumption, and cost. We describe two case studies to demonstrate modeling of an IoT environment and comparison of resource management policies. Moreover, scalability of the simulation toolkit in terms of RAM consumption and execution time is verified under different circumstances.

Received . . .

KEY WORDS: Internet of Things; IoT; Fog Computing; Edge Computing; Modelling and Simulation.

1. INTRODUCTION

The Internet of Things (IoT) paradigm promises to make ”things” including consumer electronic devices or home appliances such as medical devices, fridge, cameras, and sensors part of the Internet environment. This paradigm opens the doors to new innovations that will build novel type of interactions among things and humans and enables the realization of smart cities, infrastructures, and services for enhancing the quality of life and utilization of resources. It supports integration, transfer, and analytics of data generated by smart devices (e.g. sensors). IoT envisions a new world of connected devices and humans in which quality of life is enhanced, because management of city and its infrastructure is less cumbersome, health services are conveniently accessible, and disaster

*H. Gupta, A.V Dastjerdi, and R Buyya are with Cloud Computing and Distributed Systems (CLOUDS) Laboratory, Department of Computing and Information Systems, The University of Melbourne, Australia.
Email: {amir.vahid,rbuyya}@unimelb.edu.au,
†H. Gupta and S.K. Ghosh are with the Department of Computer Science and Engineering, Indian Institute of Technology Kharagpur, India.
Email: harshitgupta1337@gmail.com, skg@iitkgp.ac.in

recovery is more efficient. Based on bottom-up analysis for IoT applications, McKinsey estimates that the IoT has a potential economic impact of 11 trillion dollar per year by 2025— which would be equivalent to about 11 percent of the world economy. They also expect one trillion IoT devices will be deployed by 2025.

Although technologies and solutions enabling connectivity and data delivery are growing rapidly, not enough attention has been given to real-time analytics and decision making as one of the major objectives of IoT (Figure 1). Majority of current IoT data processing solutions transfer the data to cloud for processing. This is mainly because existing data analytics approaches are designed to deal with large volume of data, but not real-time data processing and dispatching. With millions of *things* generating data, transferring all of that to the cloud is neither scalable nor suitable for real-time decision making. The dynamic nature of IoT environments and its associated real-time requirements and increasing processing capacity of edge devices (entry point into provider core networks, e.g. gateways) [1] has lead to the evolution of the Fog computing paradigm. Fog computing [2] extends cloud services to the edge of networks, which results in latency reduction through geographical distribution of IoT application components, and provides support for mobile mobility.

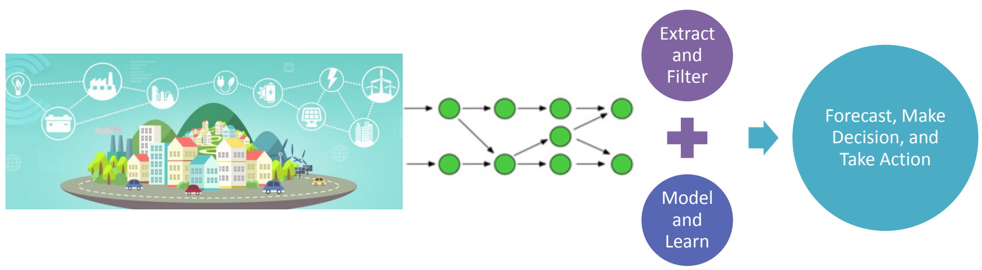

Figure 1. IoT Environment Common Objectives.

Many IoT applications (e.g. stream processing) are naturally distributed and are often embedded in an environment with numerous connected computing devices with heterogeneous capabilities. As data travels from its point of origin (e.g. sensors) towards applications deployed in Cloud virtual machines, it passes through many devices, each of which is a potential target of computation offloading. Therefore, it is important to take advantage of computational and storage capabilities of these intermediate devices. The main challenge lies in scheduling application components (operators in terms of stream processing or independent short-lived IoT tasks) in the pool of devices — from the network edge to the cloud — to meet application level Quality-of-Service (QoS) requirements such as end-to-end latency or privacy requirements while minimizing resource and energy wastage.

Resource management policies to ensure Quality of Service (QoS), avoid energy wastage, and resource fragmentation are an integral part of IoT systems. To foster innovation and development enabling real-time analytics in Fog computing, we require a an evaluation environment for exploring different resource management and scheduling techniques including operator (operator and application module have been used interchangeably in the paper) and task placement, migration and consolidation. A real IoT environment as a testbed, although desirable, in many cases is too costly and does not provide repeatable and controllable environment. To address this shortcoming, we propose a simulator called iFogSim that enables the simulation of resource management and application scheduling policies across edge and cloud resources under different scenarios and conditions.

In this paper, we discuss the architecture of iFogSim along with its design and implementation. The framework is designed in a way that makes it capable of evaluation of resource management policies applicable to Fog environments with respect to their impact on latency (timeliness), energy consumption, network congestion and operational costs. It simulates edge devices, cloud data centers, and network links to measure performance metrics. The major application model considered for iFogSim is the Sense-Process-Actuate model. In such models, sensors publish data to IoT networks, applications running on Fog devices subscribe to and process data coming from sensors, and finally insights obtained are translated to actions forwarded to actuators. In addition, we present a simple IoT simulation recipe and two case studies to demonstrate how one can model an IoT environment and plug in and compare resource management policies. Finally, we evaluate the scalability of iFogSim in terms of memory consumption and simulation execution time.

The paper is structured as follows: A formal definition of Fog computing, its concepts and benefits are presented in Section 2. Section 3 discusses the architecture of iFogSim followed by its implementation details, sample resource management policies, and a generic simulation recipe in Section 4. Case studies along with their application models and network topologies are expounded in Section 5. Section 6 presents results of scalability tests on iFogSim and compares two basic resource management policies considering metrics such as latency, energy consumption and network usage. Finally, Section 8 concludes the paper and discusses the future directions.

2. FOG COMPUTING — DEFINITION AND CONCEPTS

We define Fog computing as a distributed computing paradigm that extends the services provided by the cloud to the edge of the network. It enables seamless leveraging of cloud and edge resources along with its own infrastructure (see Figure 2). It facilitates management and programming of compute, networking and storage services between data centers and end devices. Fog computing essentially involves components of an application running both in the cloud as well as in devices between endpoints and the cloud, i.e. smart gateways and routers. Fog computing supports mobility, resource and interface heterogeneity, interplay with the cloud, and distributed data analytics to addresses requirements of applications that need low latency with a wide and dense geographical distribution. Fog computing takes advantages of both edge and cloud computing — while it benefits from edge devices' close proximity to the endpoints, it also leverages the on-demand scalability of cloud resources.

There are a number of benefits associated with Fog computing that assures its success. The first benefit is the reduction of network traffic as uncontrolled increase in network traffic may lead to

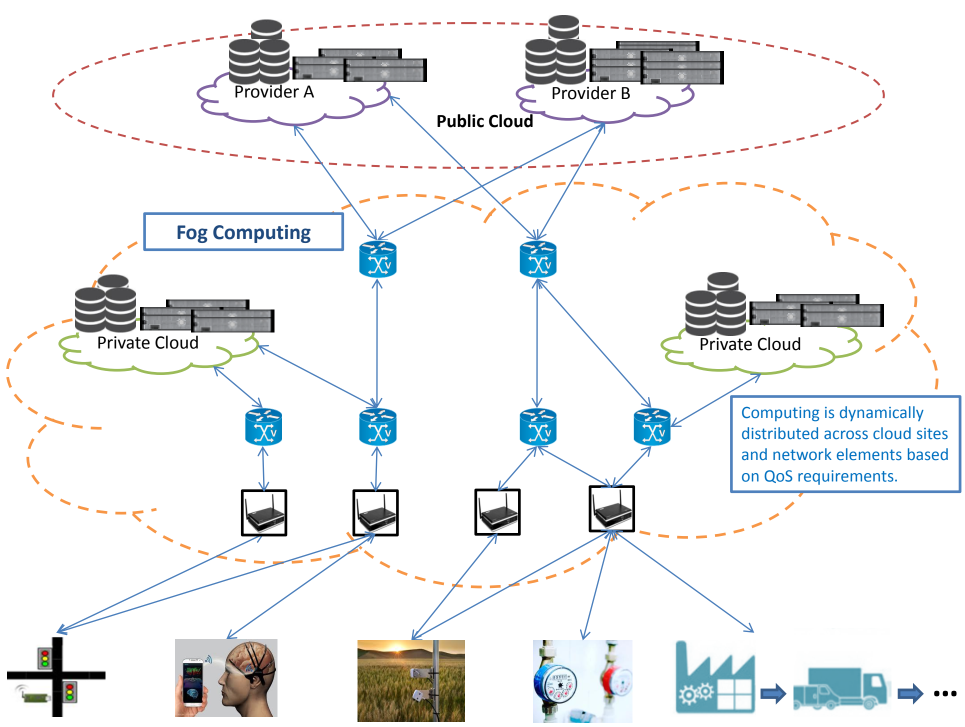

Figure 2. Distributed Data Processing in a Fog Computing Environment.

congestion, and results in increased latency. Fog computing provides a platform for filtering and analysis of the data generated by sensors by utilizing resources of edge devices. This drastically reduces the traffic being sent to the cloud by allowing the placement of filtering operators close to the source of data. Considerable reduction in propagation latency is the next important advantage of utilizing Fog computing paradigm especially for mission critical applications that require real-time data processing. Some of the best examples of such applications are cloud robotics, control of fly-by-wire aircraft, or anti-lock brakes on a vehicle. In the case of cloud robotics, the effectiveness of motion control is contingent on the timeliness of data collection by the sensors, processing of the control system and feedback to the actuators. Having the control system running on the cloud may make the sense-process-actuate loop slow or unavailable as a result of communication failures. This is where Fog computing helps by performing the processing of the control system very close to the robots — thus making real-time response possible. Finally, cloud computing paradigm, even with it's virtually infinite resources, can become a bottleneck if all the raw data generated by end devices (sensors) is sent to a centralized cloud. Fog computing is capable of filtering and processing considerable amount of incoming data on edge devices, making the data processing architecture distributed and thereby scalable.

3. ARCHITECTURE

The architecture of Fog computing environment as presented in Figure 3 involves a hierarchical arrangement of Fog nodes throughout the network between sensors and the cloud at the core of the network. In the architecture, **IoT sensors** are placed at the bottommost layer of the architecture and distributed in different geographical locations, sensing the environment, and emitting observed values to upper layers via gateways for further processing and filtering. Similarly, **IoT Actuators** operate at the bottommost layer of the architecture and are responsible for controlling a mechanism or system. Actuators are usually designed to respond to changes in environments that are captured by

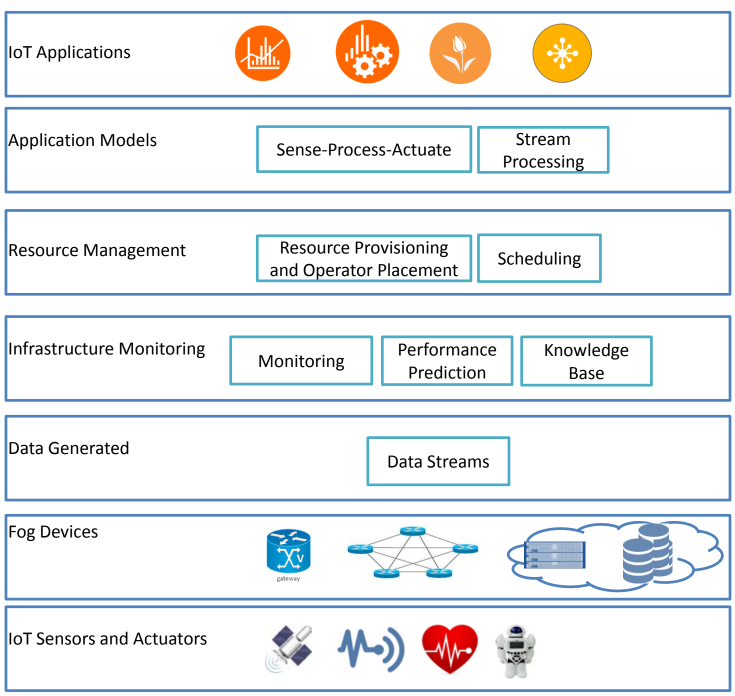

Figure 3. Fog Computing Architecture.

WSNet [13] is an event-driven simulator for wireless networks which can as well be used for IoT. It is capable of simulating nodes with different energy sources, mobility models, radio interfaces, applications, and routing protocols. Environment simulation is also supported by WSNet, in fact it offers the opportunity for modelling and simulation of physical phenomena (e.g. fire) and physical measures (e.g. temperature, humidity). These values (e.g. temperature) can be observed by the nodes, and can also impact nodes.

SimpleIoTSimulator [14] is a commercial simulator for creating IoT environments consisting of many sensors and gateway. SimpleIoTSimulator supports common IoT protocols including CoAP and MQTT as a publish/subscribe based protocol. SimpleIoTSimulator objective is enabling IoT platform and gateway vendors to improve product quality with the focus on communication protocols. Our simulators also models publish/subscribe based protocols, however our focus is on the analysis of application design and resource management policies. In addition, the SimpleIoTSimulator dose not model Fog environments where services can be deployed both on edge and cloud resources.

Since traditional wireless sensor networks and IoT simulators do not focus on modeling of large scale deployments, Giacomo et al. [15] proposed a simulation methodology for IoT systems with a large number of interconnected devices. It is designed to study low-level networking aspects. In summary, the main advantages of their approach are 1) simulation of IoT systems with geographically distributed devices; 2) simulation of are IoT devices with multiple network interfaces and protocols, as well as different mobility, network, and energy consumption models.

The OASIS standard Devices Profile for Web Services (DPWS) aims at enabling the deployment of web services on constrained devices. To accelerate the development of DPWS enabled applications, Han et al. proposed DPWSim [16], a simulation toolkit that allows developers to design, develop, and test service-based IoT applications using the DPWS technology without the presence of physical sensors and gateways.

CloudSim is developed as an extensible cloud simulation toolkit that enables modeling and simulation of cloud systems and application provisioning environments [17]. This toolkit provides both system and behaviour modeling of cloud computing components such as virtual machines (VMs), data centers, and users. However, CloudSim and other cloud environment simulators such as GloudSim [18], DCSim [19], and GroudSim [20] do not model IoT devices and stream processing applications.

In summary, although there few simulators that model IoT environments, iFogSim is uniquely designed and implemented to model Fog environment along with IoT and cloud. This enables innovation and performance evaluation of resource management policies for IoT applications such as real-time stream processing in a comprehensive end-to-end environment.

8. CONCLUSIONS AND FUTURE DIRECTIONS

Fog and Edge computing are emerging as an attractive solutions to the problem of data processing in the Internet of Things. Rather than outsourcing all operations to cloud, they also utilize devices on the edge of the network that have more processing power than the end devices and are closer to sensors, thus reducing latency and network congestion. In this paper, we introduced *iFogSim* to model and simulate IoT, Fog, and Edge computing environments. In particular, *iFogSim* allows investigation and comparison of resource management techniques based on QoS criteria such as latency under different workloads (tuple size and transmit rate). We described two case studies and demonstrated effectiveness of *iFogSim* for evaluating resource management techniques including cloud-only application module placement and a techniques that pushes applications towards edge devices when enough resources are available. Moreover, scalability of simulation is verified. Our experiment results demonstrated that *iFogSim* is capable of supporting simulations on the scale expected in the context of IoT. We also believe that the availability of our simulator will energize rapid development of innovative resource management policies in the areas of IoT and Fog computing with end-to-end modeling and simulation.

There are a number of future directions that can enhance iFogSim capabilities and resource management strategies in the context of IoT:

- **Power-Aware resource management policies:** One of the biggest challenges that most of Fog computing solutions face is how to get extra bit of battery life for Fog devices. To this end, future studies can look into new policies that dynamically and based on the battery life of devices migrate the operators. Questions such as which operator to migrate, when to migrate, and where to migrate need to be addressed in by these policies.
- **Priority-aware resource management strategies for multi-tenant environments:** Looking into scheduling policies for an environment where multiple application instances (DAGs of operators) share the same pool of resources and are assigned different Service Level Objectives (SLO) is another promising research direction.
- **Modeling failures of Fog devices:** Future research can focus on extracting failure models for the dominant failures in IoT and Fog devices. The developed models can be used to evaluate and compare reliability-aware scheduling and recovery policies for a wide range of applications.
- **Dynamic priority and SLA aware flow placement and resource scheduling (joint Edge-Network resource optimization):** In IoT environments, heterogeneous network and sensing resources have to be often shared with multiple applications or services with different and dynamic quality of service requirements. Therefore, the joint Edge-Network resource scheduling problem is another problem that we are going to investigate.
- **Modeling and comparison different virtualization techniques of IoT environments:** Future research studies can also consider and compare the performance of full virtualization, para-virtualization (as instances of hardware-level virtualization), and operating system level virtualization such as containers.

SOFTWARE AVAILABILITY

iFogSim is available for download from the CLOUDS Lab website: `http://www.cloudbus.org/cloudsim`.

ACKNOWLEDGEMENTS

This work is: (i) supported by Melbourne-Chindia Cloud Computing Research Network and (ii) initiated as part of the first author's visit to CLOUDS Lab at the University of Melbourne. It is also partially supported by the ARC Future Fellowship.

REFERENCES

1. Chang C, Srirama SN, Mass J. A middleware for discovering proximity-based service-oriented industrial internet of things. *Services Computing (SCC), 2015 IEEE International Conference on*, IEEE, 2015; 130–137.
2. Bonomi F, Milito R, Natarajan P, Zhu J. Fog computing: A platform for internet of things and analytics. *Big Data and Internet of Things: A Roadmap for Smart Environments*. Springer, 2014; 169–186.
3. Giang NK, Blackstock M, Lea R, Leung V. Developing iot applications in the fog: a distributed dataflow approach. *Internet of Things (IOT), 2015 5th International Conference on the*, IEEE, 2015; 155–162.
4. Calheiros RN, Ranjan R, Beloglazov A, De Rose CA, Buyya R. Cloudsim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. *Software: Practice and Experience* 2011; **41**(1):23–50.
5. Zao JK, Gan TT, You CK, Rodriguez Mendez SJ, Chung CE, Te Wang Y, Mullen T, Jung TP. Augmented brain computer interaction based on fog computing and linked data. *Intelligent Environments (IE), 2014 International Conference on*, IEEE, 2014; 374–377.
6. Guérout T, Monteil T, Da Costa G, Calheiros RN, Buyya R, Alexandru M. Energy-aware simulation with dvfs. *Simulation Modelling Practice and Theory* 2013; **39**:76–91.
7. Seward J, Nethercote N, Fitzhardinge J. Valgrind, an open-source memory debugger for x86-gnu/linux. *URL: http://www. ukuug. org/events/linux2002/papers/html/valgrind* 2004; .

8. Gubbi J, Buyya R, Marusic S, Palaniswami M. Internet of things (iot): A vision, architectural elements, and future directions. *Future Generation Computer Systems* 2013; **29**(7):1645–1660.
9. Stojmenovic I, Wen S, Huang X, Luan H. An overview of fog computing and its security issues. *Concurrency and Computation: Practice and Experience* 2015; .
10. Iox overview. `https://developer.cisco.com/site/iox/documents/developer-guide/?ref=overview`. (Accessed on 05/11/2016).
11. Adjih C, Baccelli E, Fleury E, Harter G, Mitton N, Noel T, Pissard-Gibollet R, Saint-Marcel F, Schreiner G, Vandaele J, *et al.*. Fit iot-lab: A large scale open experimental iot testbed. *Proceedings of the 2nd IEEE World Forum on Internet of Things (WF-IoT)*, 2015.
12. Sanchez L, Muñoz L, Galache JA, Sotres P, Santana JR, Gutierrez V, Ramdhany R, Gluhak A, Krco S, Theodoridis E, *et al.*. Smartsantander: Iot experimentation over a smart city testbed. *Computer Networks* 2014; **61**:217–238.
13. Chelius G, Fraboulet A, Fleury E. Wsnet: a modular event-driven wireless network simulator 2006; .
14. Simpleiotsimulator: The internetofthings simulator. `http://www.smplsft.com/SimpleIoTSimulator.html`. (Accessed on 05/11/2016).
15. Brambilla G, Picone M, Cirani S, Amoretti M, Zanichelli F. A simulation platform for large-scale internet of things scenarios in urban environments. *Proceedings of the First International Conference on IoT in Urban Space*, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2014; 50–55.
16. Han SN, Lee GM, Crespi N, Luong NV, Heo K, Brut M, Gatellier P. Dpwsim: A simulation toolkit for iot applications using devices profile for web services. *Internet of Things (WF-IoT), 2014 IEEE World Forum on*, 2014; 544–547.
17. Calheiros RN, Ranjan R, Beloglazov A, De Rose CA, Buyya R. CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. *Software: Practice and Experience* 2011; **41**(1):23–50.
18. Di S, Cappello F. GloudSim: google trace based cloud simulator with virtual machines. *Software: Practice and Experience* 2014; .
19. Tighe M, Keller G, Bauer M, Lutfiyya H. DCSim: a data centre simulation tool for evaluating dynamic virtualized resource management. *Proceedings of the 2012 8th international conference on Network and service management (cnsm) and 2012 workshop on systems virtualiztion management (svm)*, 2012; 385–392.
20. Ostermann S, Plankensteiner K, Prodan R, Fahringer T. GroudSim: an event-based simulation framework for computational grids and clouds. *Proceedings of the 2010 Conference on Parallel Processing*, Euro-Par 2010, Springer-Verlag: Berlin, Heidelberg, 2011; 305–313.