

Urban Computing Experiment by Mixing Fog Computing Simulation and Public Open Street Map Data

Lucas Alberto S. Santos, Claudio F. R. Geyer
Instituto de Informatica
Universidade Federal do Rio Grande do Sul (UFRGS)
Grupo de Processamento Paralelo e Distribuido
Porto Alegre, RS, Brasil
lucasa@softwarelivre.org, geyer@inf.ufrgs.br

Abstract

Fog Computing has attracted the attention of researchers in the field of Large Distributed Systems for its ability to handle the limitations of the cloud computing paradigm when applied to the reality of IoT and its massive amount of distributed sensors and wireless mobile devices at the edges of the network. Simulation of Fog computing makes possible to experiment with complex distributed processing scenarios at a very low cost compared to creating real testbeds.

In this work we extend the iFogSim Fog Computing simulation framework to more appropriate simulate a smart city experiment. We developed a experiment to help locating citys stolen cars where fog devices distributed on the city's electronic radars capture images that are processed to detect stolen car's plates. The public open GIS data of the colaborative Open Street Map project serve as a reliable basis to support Urban Computing research, this work contributes to the field as an exercise of urban computing simulation integrated with open urban data.

1. Introduction

Urban Computing seeks to optimize the processes of life at cities by intelligently acting on the massive data generated by various local sources such as people, organizations, houses, buildings, sensors, devices, vehicles, etc. This area of research in Intelligent Cities aims to analyze a large amount of heterogeneous data to extract solutions to address critical problems for cities such as pollution, energy consumption and congestion. This localized computing model connects urban sensing, data management, data analysis, and service delivery in a recurring process of continuous improvement of people's quality of life, urban control systems, and the environment [2].

According to [7], the Internet of Things can act, as well as other environments, in cities as well. The application of the IoT paradigm in the urban universe is of great interest to public managers who seek to make better use of state resources, advancing the quality of services offered to citizens while reducing the operational costs of public administration.

2. Fog Computing

Recently, has become popular in the research area of distributed systems, the Fog Computing model to deal with the limitations of the Cloud Computing paradigm. The centralized model of the remote public data center does not scale when applied to thousands of sensors geographically distributed and connected to intelligent things at the edges of the internet [2]. Fog Computing bridges the gap in the Internet for Things (IoT) with a proposed computing and communication architecture that distributes computing, control, and storage functions to near end-user devices by introducing new devices (fog nodes) and ad-hoc networks between the central cloud and users at the edge of the Internet [5].

Researching in the field of urban computing are very laborious and costly to carry out real-scale tests, due to the complexity of this intrinsically interdisciplinary theme, the number of actors involved and the high scale of urban processes. Simulating tools and techniques are quite popular in the literature for experimental studies on distributed computing architectures. The simulation anticipates the behavior of the virtual components, helps researchers to understand and improve their modeling more efficiently [6]. For the simulation of fog computing scenarios, was recently published the iFogSim tool [4] based on the CloudSim platform [3], a popular Java framework for Discrete Event Simulation. The iFogSim framework extends the CloudSim platform adding fog computing entities such as Sensor, Actuator, Fog Device, Cloud, etc.

3. Extending the iFogSim Framework

The iFogSim architecture has been extended to include extra information properties to the entity classes of the framework. The *FogDevice* class has been extended into a new *SmartFogDevice* class to better support geographic properties and spatial range information of wireless network radio signal.

Our implementation is a prototype of a Urban Computing simulator because it incorporates spatial local context to the fog computing simulation engine. We implemented a function to import geographic locations from a *GeoJSON* format file, which is a standard format for exchanging geographic data collections.

3.1. Experiment - Locating Stolen Cars

The source code of iFogSim includes a demo source code "Case Study 2 - Intelligent Surveillance through Distributed Camera Networks". This scenario described by the authors follows the *Sense-Process-Actuate* model, where several surveillance cameras links to a network of fog devices cooperating with the Cloud to process machine learning computation task over massive data from sensors.

In this work we implemented an experiment inspired by the *Sense-Process-Actuate* scenario proposed at the iFogSim original paper. In our experiment, image sensors are located in the sixty four electronic radars of the city of Porto Alegre - RS. When a car is photographed by a radar, a plate recognition program runs over the image to locate plate signs of recently robbed cars. If car plate is identified, a notification message is then sent to the mobile device of the end-user, located at the Polices Palace. The cloud data center is located at the headquarters of the Municipal Data Processing Company (Procempa).

The modular application executed by the fog network has the following data flow:

- The *Motion Capture and Detection* module takes periodic photos of the traffic path as the tuples are routed if there is a perception of a moving car in the image.
- The *Plate Recognition* module looks for suspicious plates in the image. This module applies a machine learning algorithm type with high computational demand.
- If a stolen car is detected, the image is immediately forwarded to a *Post-Processing* module that envelops the suspect image and forwards it to the end-user device, including additional metadata from where and when the visual record was taken.

The main technical aspects of simulation configuration are:

- Each of the 64 electronic radars of Porto Alegre is a local Cyber-Physical System (CPS) composed by:
 - Sensor: 1 camera 720p que captura 2 frames/seg.
 - Fog/edge device: 1 ARM hardware - Raspberry Pi 3, clock 1 GHz and 1 GB RAM.
- Cloud has 16 CPUs of 3 GHz and total of 24 GB RAM.
- The Wifi Routers are connected to the Cloud by a link of 10 MByte/seg.
- The Radars edge devices connects to Wifi Routers that are in their radio signal range (local neighbourhood). We implemented an algorithm to optimize the distribution of wireless network routers in the city, to promote that a same wifi point could be shared between multiples geographically closed Raspberry Pi devices.
- The Module Plate Recognition has an output tuple rate of 0.5% forwarded, so only a small amount of the plate images will be given as positively identified and then forwarded to the mobile device of the End User.

This smart city simulation scenario was organized with iFogSim's components abstractions of *Sensor*, *Actuator*, *Application*, *Network Links*, *Fog Devices* and *Cloud*. The georeferenced open data information of the traffic lights and radars of the City of Porto Alegre were downloaded from the Overpass API, a Open Street Map (OSM) webservice client [1]. There are many open source tools to convert the OSM XML native format to GeoJSON.

4. Smart City Simulation Results

The experiments were performed with 30 simulation execution, each execution performs 1 hour of simulation time, for each of the scenarios:

- a) execution machine learning by the fog devices;
- b) processing centralized in the cloud.

The results on the **Table 1** are the averaged values of the 30 executions.

Type	Network (MB)	Power Consumed (W)	Latency (ms)
At Cloud	362324,13	37773570,58	222,57
At Fog Devices	6427,26	37404648,77	274,91

Table 1. Results of the Simulation Scenarios

The result of analysis of the information collected from simulations show that:

- The scenario of intensive processing in the cloud floods the WAN network with image files, taking the network bandwidth as a bottleneck.

- At the distributed processing scenario, the images files are processed locally, so only images with detected cars are sent as successful notifications.
- The *energy consumption* is almost similar between the scenarios, indicating that the low-power computing hardware of the edge devices consumes almost the same as the Cloud to process the work. This experiments were copied from the *hardware characteristics* published at the iFogSim paper. We are going to review and calibrate the energy consumption and MIPS rates of the simulated hardware, to get more realistic results.
- The average *latency time* (delay) between the car photo capture and the end-user notification was lower for the cloud scenario, indicating a expected behaviour that the simulated cloud of 16 3GHz CPUs is more powerful than the processing power of low power ARM devices.

The iFogSim simulator makes possible to collect several useful information for simulated scenario analysis and comparison, below are the information used to compare the scenarios:

- Total simulation time;
- Total memory used in the simulation;
- Total network bandwidth consumed;
- Total energy consumed (cloud + fog + networking);
- Latency between image capture and notification to the end user;

We implemented a interactive graphical UI for visualization of the urban map (Figures 1, 2 and 3) that shows the dynamics of data between the application modules that execute on the fog devices, along the passage of a virtual simulation time. The simulation information is designed as animated overlays on the actual city map, making it possible to direct focus to local information or even broadening the observation to see large urban areas.

The source code of this work with instructions for experiment reproduction can be accessed at ["http://github.com/lucasa/ismartsim_project"](http://github.com/lucasa/ismartsim_project).

5. Future Work

The Java abstractions of the iFogSim framework for the Fog Computing's common entities serve as the basis for creation of a high-level architecture for simulation of smart city scenarios and their heterogeneity of IoT devices, Wifi networks, fiber optics, ad-hoc bluetooth links, urban mobility, software off-loading, etc. We are going to explore others more complex fog computing scenarios and implement

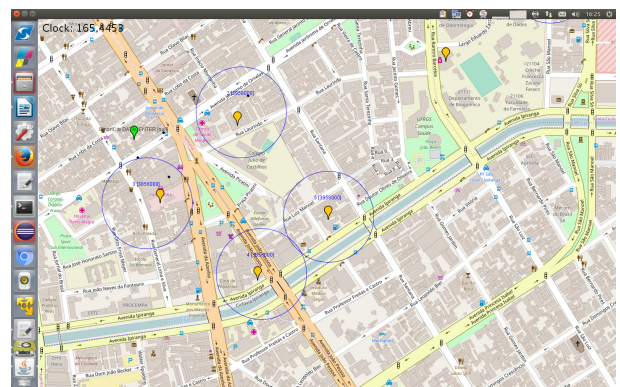


Figure 1. Screenshot of the simulation map showing the location of the fog devices (yellow marker) and the coverage radio range of the wireless network of the attached local router (blue limits).

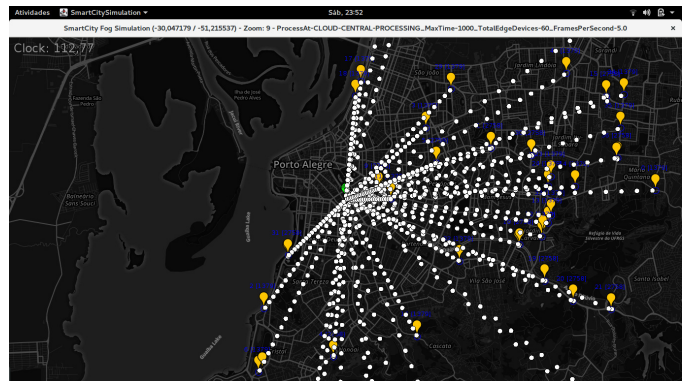


Figure 2. Animation of recorded events log: streams of tuples (raw image files) going through WAN network from the edges to be processed at central cloud.

new simulated abstractions for hardware and software components of complex smart city systems.

The use of public open data sources is a very interesting topic for smart city interdisciplinary research, going further Open Street Maps GIS data to get the simulation more realistic by using updated raw data from auditable sources.

The user interface for editing and replaying can be rewrite as web component to run in a internet browser, opening the smart city simulation area to the world of creative Javascript and advanced usability studies. As a online tool, we get platform that potentially allows collabora-

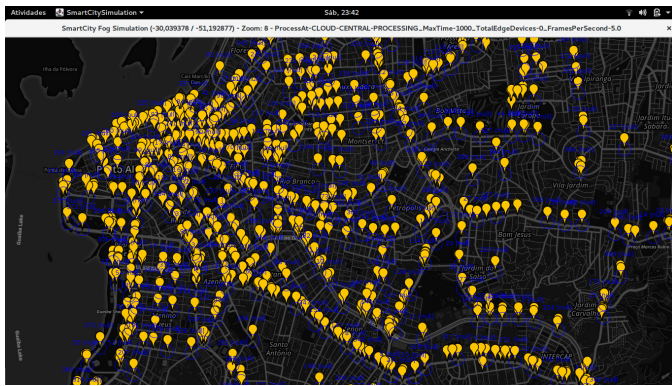


Figure 3. Visualization of a hypothetical scenario of simulation (not executed in this work) with sensors and Wifi networks distributed by all the traffic lights of Porto Alegre.

tive work in a same simulation project.

6. Conclusions

Fog Computing have been attracting the attention of researchers in the field of Large Distributed Systems for its ability to handle the limitations of the cloud computing paradigm in front of a new paradigm of location-aware data abundance by small massive distributed IoT sensors and low-power mobile computing hardwares at the edges of the network. There are still few simulators like iFogSim for the development of fog computing experiments.

In this work we implemented a support for import and visualization of georeferenced information in the backend of the simulation engine IFogSim/CloudSim. Thus, the potentialities of the iFogSim framework for cloud computing simulation become more useful for simulating Urban Computing in smart city scenarios. The experiment implemented on the iFogSim framework, although quite simple, was useful for study a variety of configuration parameters and events generated by the simulation engine.

The simulation of Smart City experiments, when uncomplicated and high level, opens a whole space for multidisciplinary academic research aimed at the application of technologies of computation and communication to solve current and future problems in the big cities of the world.

References

[1] Openstreetmap community: Osms overpass application. Accessed April 4, 2010.

- [2] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli. Fog computing and its role in the internet of things. In *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*, pages 13–16. ACM, 2012.
- [3] R. Buyya, R. Ranjan, and R. N. Calheiros. Modeling and simulation of scalable cloud computing environments and the cloudsims toolkit: Challenges and opportunities. In *High Performance Computing & Simulation, 2009. HPCS'09. International Conference on*, pages 1–11. IEEE, 2009.
- [4] H. Gupta, A. Vahid Dastjerdi, S. K. Ghosh, and R. Buyya. ifogsim: A toolkit for modeling and simulation of resource management techniques in the internet of things, edge and fog computing environments. *Software: Practice and Experience*, 47(9):1275–1296, 2017.
- [5] R. Mahmud, R. Kotagiri, and R. Buyya. Fog computing: A taxonomy, survey and future directions. In *Internet of everything*, pages 103–130. Springer, Singapore, 2016.
- [6] A. Sulistio, C. S. Yeo, and R. Buyya. A taxonomy of computer-based simulations and its mapping to parallel and distributed systems simulation tools. *Software: Practice and Experience*, 34(7):653–673, 2004.
- [7] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi. Internet of things for smart cities. *IEEE Internet of Things journal*, 1(1):22–32, 2014.