Demo: DingNet: A Simulator for Large-Scale IoT Systems with Mobile Devices

Michiel Provoost
Department of Computer Science, Kulak
Katholieke Universiteit Leuven
3001 Leuven, Belgium
michiel.provoost1@student.kuleuven.be

Danny Weyns
Department of Computer Science, Kulak
Katholieke Universiteit Leuven
3001 Leuven, Belgium
danny.weyns@kuleuven.be

Abstract

The Internet-of-Things (IoT) provides an enabling layer for smart cities. However, engineering such large-scale IoT deployments is challenging since these systems typically consist of complex distributed infrastructure of gateways and devices. To investigate and test IoT solutions before deployment in the field, engineers can make use of simulation. In our research, we study large-scale IoT systems for smart city applications that comprise of mobile devices that sense parameters of the environment and send the sensor data to a user application via stationary gateways. As existing IoT simulators are not sufficient to support our research, we developed a novel simulator of DingNet, an IoT system that is deployed in Leuven. In this paper, we give an overview of the simulator design and we illustrate how we used for the study of an IoT scenario with multiple mobile devices.

1 Introduction

The introduction of smart technologies is reshaping our cities rapidly. Smart city applications, such as smart parking systems and area surveillance are deployed on large-scale Internet-of-Things (IoT) systems. However, engineering such IoT deployments is challenging since these systems typically consist of complex infrastructure of gateways and devices that are spread over the city area. To investigate, design, implement, and test IoT solutions before actual deployment in the field, engineers can make use of simulation.

In our research, we study large-scale IoT systems for smart city applications that comprise of potentially a large number of devices that collect data and send that to gateways that are deployed in the city. Our focus is on systems with mobile sensor devices that move through heterogeneous areas of a city. To support our research, a simulator is required that supports the following objectives: (i) large-scale IoT deployments, (ii) mobile devices, and (iii) fine grained environment modeling, (iv) directly mapping to a real IoT network. A number of simulators have been developed to support researchers and engineers of IoT systems, e.g., [1,2]. However, to the best of our knowledge, none of the existing simulators supports all the required objectives. Therefor, we developed a novel simulator for IoT that supports researchers and engineers of large-scale deployments with mobile devices. This simulator directly maps to the DingNet IoT network that is deployed in the Leuven area in Belgium. An advantage of the simulator is that the software tested on DingNet can directly be deployed on the DingNet infrastructure.

The remainder of this paper is structured as follows. In Section 2, we elaborate on the requirements for the IoT simulator and we explain how we used these to design the simulator. Section 3 explains how the simulator can be used to set up an experiment. Finally, in Section 4, we draw conclusions and look at planned work.

2 Simulator

We start with introducing DingNet. Then we outline the requirements for the DingNet simulator. Finally, we explain the design of the simulator and highlight its main features.

2.1 DingNet

DingNet[1] is a LoRaWAN-based IoT network that covers the city of Leuven. The network contains 14 gateways that are deployed in high buildings throughout the city to ensure good coverage. LoRaWAN networks are characterized by large communication range. Devices that communicate with gateways within communication range require low power consumption. The data can be encoded using keys to ensure secure transmission. However, data traffic is limited to a few 100 packets of a 10's of bytes a day per device. The gateways forward the data that they receive from the devices to an application server where the data is further processed by a user application. This communication uses the MQTT protocol.[2] Usually battery-powered (or energy-harvesting) IoT devices are used that can sense environment parameters and send the data to gateways. Downstream traffic from the gateways to devices is also possible. This traffic enables adaptation of the settings of devices. Typical applications for DingNet are garbage bins that send signals when they are full, devices that the air quality in the city area, etc.

[1] https://admin.kuleuven.be/icts/english/services/dingnet
[2] http://mqtt.org/

International Conference on Embedded Wireless
Systems and Networks (EWSN) 2019
25–27 February, Beijing, China

ISBN: 978-0-9949886-3-8

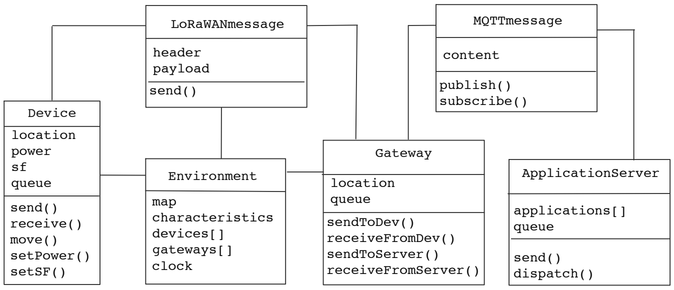

Figure 1. Design simulator (excl. GUI and Analysis)

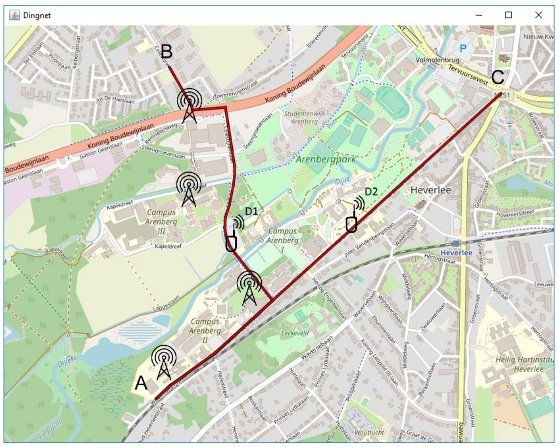

Figure 2. Part of the map of Leuven used in the simulator

2.2 Simulator Requirements

To support researchers and engineers of large-scale IoT systems with mobile devices, we identified the following requirements for the simulator based on the objectives outlined above: R1: support the three tiers of IoT deployments: devices, gateways, applications; R2: support secure communication using the DingNet protocols between the tiers; R3: support environment modeling, incl., movements, time, message transmission, collisions, and interference; and R4: Support specification of IoT configurations and scenarios.

2.3 Simulator Design

Figure 1 shows the high-level design of the DingNet simulator. *Device* represents a mobile sensor device with a location, message queue, power setting, and spreading factor (sf); the last two can be adapted. A device can send a *LoRaWANmessage* with a header and payload to a *Gateway* that has a location and message queue. A gateway can send a LoRaWAN message to a device; it can also send a *MQTT message* with content to one of the *applications* at the *ApplicationServer* using a publish/subscribe mechanism. The *Environment* has a map with devices and gateways located on it, a clock, and set of characteristics, incl. path loss and shadow fading that allow modeling different transmission characteristics in the environment. The simulator is freely available.[3]

3 Simulator at Work

We now illustrate how we used the simulator for an IoT scenario with two mobile devices. Figure 2 shows the part

[3]https://people.cs.kuleuven.be/danny.weyns/software/DingNet/index.htm

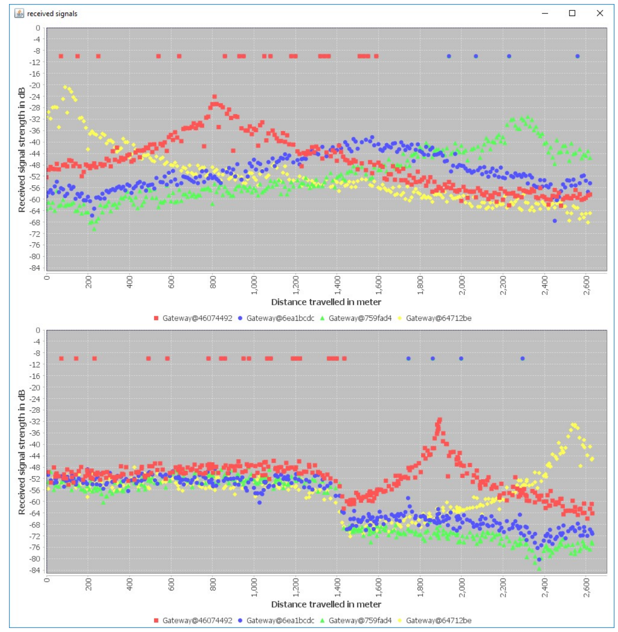

Figure 3. Signals received by gateways and collisions

of the map of Leuven used in the experiment. The antenna symbols represent gateways that are placed at high buildings. The wireless sensor symbols represent mobile devices.

We consider a scenario where devices track the air quality while moving through areas of the city with different levels of network interference. Device D1 moves from A to B; D2 from C to A. D1 sends one data sample every 10 meter at maximum settings for power and spreading factor. D2 sends a message at a random interval between 1 and 15 meter.

Figure 3 (top) shows the strength of the signals of D1 received by the different gateways. The line at the top shows collisions of messages. Figure 3 (bottom) shows similar results for D2. Notice that there is more interference in some areas (e.g., from 1300m to 1700m for D2) resulting in lower received signal strength. Overall, 93.0% of the messages sent by D1 arrived; the rest was lost due to collisions. For device D2, 94.4% of the messages arrived.

4 Conclusions and Outlook

We presented a novel simulator called DingNet that maps to a IoT system deployed in Leuven. The simulator supports researchers and engineers to design and analyze large-scale IoT systems with mobile devices. We illustrated how we used the simulator to study the behavior of a scenario with multiple mobile motes. In future research, we plan: (i) to enhance the simulator with extra features such as support for adding feedback loops to realize self-adaptation [3], and (ii) to collect data from field experiments and use this to bring the simulator as close as possible to the real setting.

5 References

[1] M. C. Bor, U. Roedig, T. Voigt, and J. M. Alonso. Do LoRa Low-Power Wide-Area Networks Scale? MSWiM 2016

[2] B. Reynders, Q. Wang, and S. Pollin, A LoRaWAN module for ns-3: implementation and evaluation, WNS3 2018

[3] D. Weyns, Software Engineering of Self-Adaptive Systems, Handbook of Software Engineering, Springer 2019

6 Demo Requirements

There are no special requirements to present the demo, but a larger screen that can be connected to the laptop of the presenter to show the demo would be appreciated.