Simulation of mobile edge-cloud applications using Mininet-WiFi

Tiago Castanheira, Joaquim Silva,
Eduardo R. B. Marques, and Luís M. B. Lopes

CRACS/INESC-TEC & Faculty of Science, University of Porto, Portugal

Abstract. The ubiquity of mobile devices led to the consideration of mobile edge clouds (MECs), networks formed by devices in close proximity without infrastructural support, or at most optionally complemented by cloudlets, lightweight servers serving small areas. MECs empower novel crowd-sourcing, proximity-aware applications, where computation and communication mostly occur at the edge of network, without depending on infrastructural communications and cloud services. Simulation approaches are required to validate MEC applications reliably and at scale, as field experiments can be onerous and hard to replicate, in regard to factors such as the number of nodes, geographical area, network setup, or mobility patterns. We outline our approach for the simulation of MEC applications making use of the Mininet-WiFi platform, taking into account real-world application scenarios from the Hyrax project.

1 Introduction

Personal mobile devices like smartphones or tablets have become ubiquitous in recent years, and have become relatively powerful machines in terms of storage, memory, built-in sensors and networking capabilities. Many mobile applications are designed in line with the mobile cloud computing paradigm, where computation and storage rely on centralized cloud servers and infrastructural communications. Novel proximity-aware, crowd-sourced applications, however, challenge this approach, by requiring computation, communication and storage to occur primarily at the edge of the network. Mobile edge-clouds (MEC), where nearby devices form ad-hoc networks using device-to-device (D2D) communication, possibly supported by servers on the edge of the network known as cloudlets, provide a suitable approach for these applications.

In the Hyrax project (`http://hyrax.dcc.fc.up.pt`), we designed and experimented with some significant MEC applications and services, and have been developing a communications middleware in their support [3]. Synthetic experiments in a lab setting or field experiments in real-world setting may allow us to evaluate the effectiveness of our efforts and gain insight on the design of MEC applications, but only to a limited extent. The flexible evaluation of MEC applications through simulation may provide a significantly broader understanding of key factors and their impact at scale, for instance user mobility, choice of network architecture, different communication technologies, or application parameters. We next discuss an Hyrax application that provides a motivating use-case for simulation, then describe our ongoing work in developing a MEC simulation framework based on the use of the Mininet-WiFi platform [1].

2 Case-study application

The User-Generated Replays (UGR) is a major case-study application of the Hyrax project [4, 5], providing a good motivation for the validation of MEC applications through simulation. The UGR application allows users to capture and share videos in a crowded venue, e.g., during a sport event. Traditional applications rely on infrastructural communications for data dissemination, leading to traffic congestion, e.g., infrastructural Wifi and 3G/4G access can often be quite slow in crowded venues. UGR mitigates this problem by leveraging edge-cloud networks formed by nearby devices and/or proximity cloudlets, using infrastructural communications as a last resort.

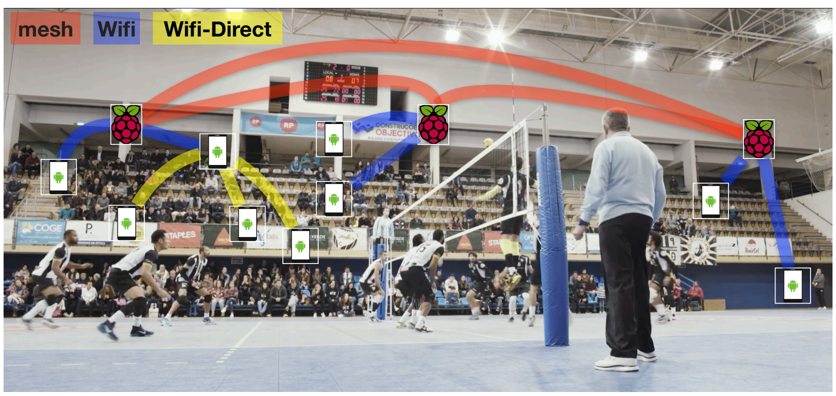

Fig. 1. Field experiment for the UGR scenario (picture from [4]).

Figure 1 depicts a photo of a UGR field experiment we conducted recently in the real-world setting of a Portuguese volleyball league game [4]. The picture also shows a schematic illustration of a hybrid edge-cloud architecture supporting the UGR functionality during the experiment. Lightweight cloudlets enabled through Raspberry Pi devices were deployed at different locations in the game venue. Each cloudlets served as WiFi access points and hosted a video storage service, allowing devices to upload and download videos from it. Cloudlets were also connected among themselves through a mesh network, such that videos uploaded by devices to a particular cloudlet would be automatically disseminated to the remaining cloudlets through the mesh network, thus allowing users at different locations to share videos almost in real-time. Meanwhile, mobile devices could spontaneously form WiFi-Direct groups, that would act as video caches, i.e., download requests for videos available in a group would be served locally bypassing the cloudlet layer.

Real-world experiments such as this are of course impossible to reproduce under the same exact conditions. Moreover, they are onerous to execute in logistical terms, and inflexible in the sense of requiring a configuration defined beforehand that may in any case be hard to control in fine-grained manner. Simulation is the obvious approach for evaluating MEC applications with reproducible behavior and under different configurations. In the UGR example scenario, we have the following key ingredients:

- Architectural design choices: How will the UGR system behave using cloudlets but no Wifi-Direct groups? Or merely devices connected through WiFi access points? Or even strictly just mobile devices connected through WiFi-Direct groups?
- Scalability: What is the impact of increasing the number of users, the rate of video uploads and downloads, or the size of videos? In a nutshell, how does the system scale?
- Mobility: How do mobility patterns affect WiFi-Direct group formation and the effectiveness video caching within them? Is mobility disruptive in the sense of inducing churn and instability, or, on the other hand, can it have a positive influence by aiding video dissemination?

3 MEC applications in Mininet-WiFi

Mininet-WiFi [1] is an extension of Mininet [2] for mobile wireless networks. Mininet supports the creation of virtual networks with an arbitrary number of hosts within a single Linux host. Through a lightweight virtualization scheme, Mininet provides the illusion of multiple hosts that may run unmodified Linux binaries/utilities and use the standard Linux networking stack. For scripting, the Mininet functionality is exposed through a Python API. Building upon Mininet, Mininet-WiFi adds support for various types of WiFi communications (standard WiFi, WiFi-Direct, mesh networking, etc) and different types of wireless nodes (mobile stations, static hosts, or access points). Moreover, it provides support for the definition of mobility patterns and emulation of the wireless physical medium and associated signal propagation models. A Mininet-WiFi screenshot is shown in Figure 2, illustrating a particular simulation with access points and mobile stations deployed over a certain physical area.

We have been using Mininet-WiFi for the simulation of MEC scenarios that are very similar to the UGR application we described. An initial configuration defines aspects such as the number of nodes and their types, reflecting a choice of a network architecture in line with the variations we discussed earlier (e.g., enabling/disabling cloudlets, WiFi-Direct groups, access points), the dimensions of the physical space and the pattern that applies to the movement of mobile stations over time, parameters for tuning WiFi-Direct group formation (if is enabled). and the positions of cloudlets (modeled as static hosts) and access points. As the simulation begins, mobile station nodes start roaming in the virtual space and generating contents (plain binary files) with a certain rate over time following a Poisson distribution. When stations connect the same WiFi access point, or alternately, they are in physical proximity and form a WiFi-Direct group,

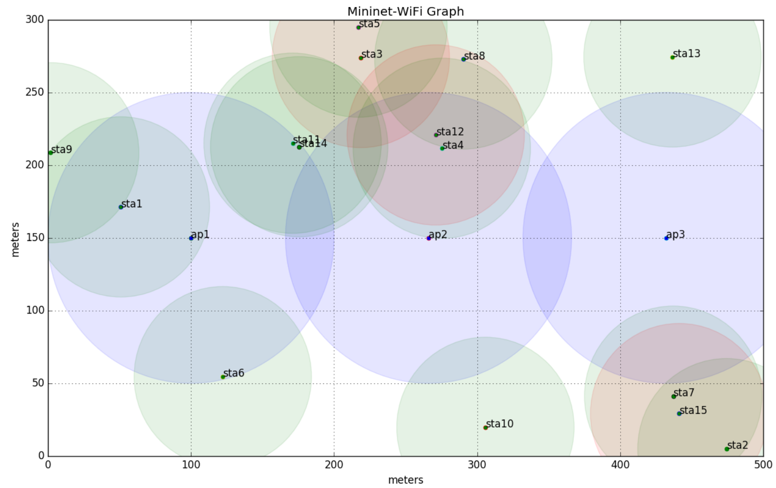

Fig. 2. Mininet-WiFi screenshot.

they synchronize their contents (while their connection lasts) following a gossip scheme for data dissemination. The same process of synchronization occurs between stations and cloudlets or among cloudlets.

Currently, we are ready to start perform a systematic, large scale evaluation of this overall scenario. We expect to present the main results and conclusions at INFORUM, if the abstract is accepted for presentation.

References

1. Fontes, R.R., Afzal, S., Brito, S.H., Santos, M.A., Rothenberg, C.E.: Mininet-wifi: Emulating software-defined wireless networks. In: Proc. 11th International Conference on Network and Service Management (CNSM). pp. 384–389. IEEE (2015)
2. Lantz, B., Heller, B., McKeown, N.: A network in a laptop: rapid prototyping for software-defined networks. In: Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks. p. 19. ACM (2010)
3. Rodrigues, J., Marques, E.R.B., Lopes, L., Silva, F.: Towards a middleware for mobile edge-cloud applications. In: Proc. 2nd Workshop on Middleware for Edge Clouds & Cloudlets (MECC'17). pp. 1:1–1:6. MECC'17, ACM (2017)
4. Rodrigues, J., Marques, E.R.B., Silva, J., Lopes, L., Silva, F.: Video dissemination in untethered edge-clouds: a case study. In: Proc. 18th IFIP International Conference on Distributed Applications and Interoperable Systems (DAIS). pp. 137–152. DAIS'18, Springer (2018)
5. Silva, P.M.P., Rodrigues, J., Silva, J., Martins, R., Lopes, L., Silva, F.: Using edge-clouds to reduce load on traditional wifi infrastructures and improve quality of experience. In: Proc. 1st IEEE International Conference on Fog and Edge Computing (ICFEC). pp. 61–67. ICFEC'17, IEEE (2017)

Acknowledgements. This work was funded by the NORTE 2020 project NORTE-01-0145-FEDER-000020, the COMPETE 2020 Programme project POCI-01-0145-FEDER-006961, and by FCT projects CMUP-ERI/FIA/0048/2013 and UID/EEA/50014/2013.