Performance Evaluation of Round Robin CQI based Cellular Network

Università di Pisa

Antonio Le Caldare, Vincenzo Consales, Vincent Della Corte

year 2017/2018

Contents

Chapter 1

Modelling

1.1 Introduction

In these paragraph we describe how we modeled the Celluar Network described in the specifications.

- **A Web Server**, which generates data in the form of packets to be trasmitted to users. For our pourposes we have defined the class `UserPackets` which includes a `start_time` field and its interface (named `UserPacket_m`) includes a getter/setter method to update this field.

  The size of each packet is a RV with a uniform distribution, since the service demand has to be uniform. Moreover the packet interarrival time to the antenna has to be an exponential RV, so each packet is generated properly to satisfy this requirements.

- **An Antenna**, which has FIFO queue for each user. Packets received form **Web Servers** are stored inside queues and then are sent in a unicast way according to the Round Robin policy (which is described in the next section).

- **A Mobile Station**, which personifies a generic user connected to the antenna. On each timeslot it sends a channel quality indicator (CQI), which is a number between 1 and 15 that define the number of bytes the antenna can pack into a Resource Block (RB).

  CQIs are integer RVs generated according the following scenarios:

  1. Uniform, each user generates a RV $\sim U(1, 15)$
  2. Binomial, each user generates a RV $\sim Bin(n, p_i)$, where n is the number of users, and $0 < p_i < 1$ depends on the user i.

To build our model and to run rimulations we used the framework **OMNeT++ v5**, so each item described before is defined by a `*.ned` file. Each `Mobile Station` computes some statistics: slotted throughput (related to each time slot) and response time of received packets. The `Antenna` compute also statistics about the frame filling, which we will describe later.

The `CellularNetwork.ned` file shows how the previous modules are connected to obtain the network. Since frames are sent in a unicast way, there are multiple instances of the Web Server module, one for each Mobile Station, seeded in a different way in order to have IID RV.

The obtained network, by setting $n = 10$, is the following:

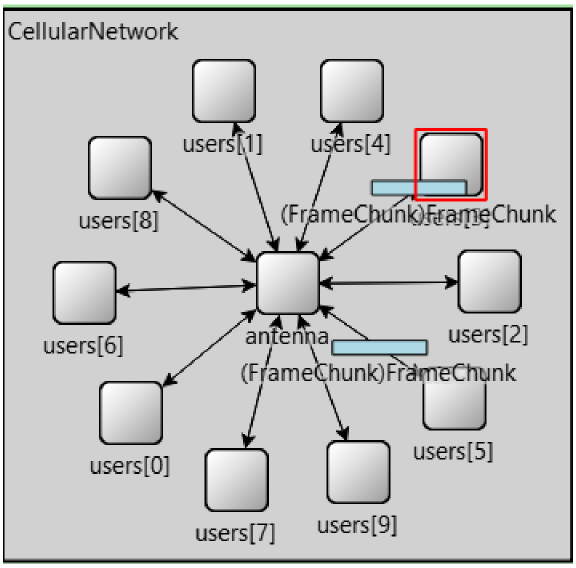

Figure 1.1: Simulated Network (omnet++)

1.2 Frame Chunks

Packets are delivered to users by enveloping that inside RBs. As requested by specifications $RBsize_{\mathrm{i,t_j}} = f(CQI_{\mathrm{i,t_j}})$ where i is index of *user*, and t_j is index of *timeslot*. To do that, the scheduler receives CQIs from the `Mobile Stations` at beginnig of each time slot, then compute every $RBsize_{\mathrm{i,t_j}}$ and fills the frame according to scheduling policy. Note that a frame can carry RBs for different users so, generally speaking, a frame in a specific time slot t_j can have RBs of different size. To cope these requirements we defined a new module `Frame Chunk` wich groups all RBs addressed to a specific `Mobile Station`. By introducing this module we can consider a frame as a collection of `Frame Chunks`. To deliver the whole frame we must send each `Frame Chunk` to its user in a unicast way.

1.3 Schedulers

In this section we will analyze the frame filling policy wich are defined in the module `Scheduler`. Let's consider a user $0 \leq i \leq n$. Starting from user i, the scheduler allocates a new `Frame Chunk` and fills it with packets taken from the $\texttt{FIFOQueue}_{\texttt{i}}$. A `Frame Chunk` is considered full if it contains 25 RBs because it corresponds to the whole frame. If the frame is not full the scheduler must allocate others `Frame Chunks` in order to fill the residual space using one of the two following policies.

1.3.1 Round-Robin Frame Fill

1.3.2 Best CQI based Frame Fill

Chapter 2

Results Analysis

2.1 Scenarios