Wireless & The IoT

Lab 4: To Cell or not to Cell? Closing the Loop on the LPWAN Landscape

Introduction

The purpose of today's ~~lab~~ design assignment is to round out our evaluation of design options in the LPWAN space with the competiting ISM options. It's a bit less lab-y #ThanksCOVID, but hopefully will be an interesting exercise.

The Assignment

We are going to stitch together your work over the last two weeks to form a comprehensive look at the tradeoffs for individual and (global?) fleet-wide deployment of an IoT system. Recall the major metrics:

- A 6-month pilot of 1,000 devices.
- Each device will send a 1 kB request once every 15 minutes, 24 hours a day.
- Let us also assume that once per month, you will update device firmware (a 10 MB operation).

Let us also add device context: Let's say your device is traffic monitor, which is counting pedestrians, bike, vehicles, and other flow at intersections. That is, a new constraint:

- The target density of your device is 1 at each street intersection.

We are going to look at the comparative coverage, economic cost, and energy cost of the various means you could use to network your deployment.

What to submit?

Please use this document as a template, add your responses directly, and export it as a PDF to Gradescope. This is a hard problem, with lots of subtle details. Folks are encouraged to collaborate as much as you like with others. If you work with others, please put everyone's name who worked together below. I believe I have also configured Gradescope to allow "group submission," so please submit to Gradescope as a group.

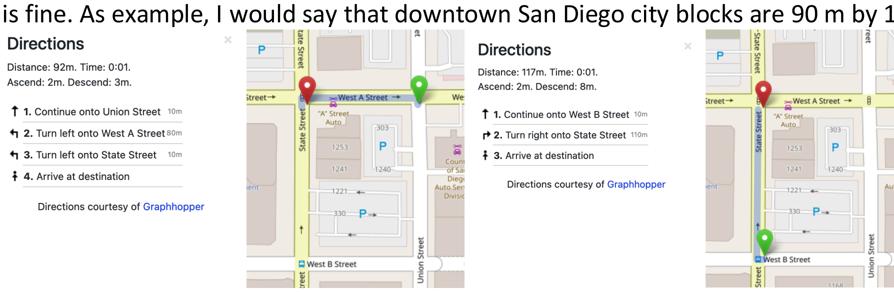
____

(your name**(S)** here)

“Q”0: Make some decisions

Choose a non-cellular LPWAN provider (i.e. an entity providing wide-area LoRa coverage [such as Helium, or The Things Network, or Senet, or…], Sigfox[1], a TV Whitespace user (if you can find one?), etc). If you’re unsure, Helium is a reasonable default choice.

Pick one of the countries you looked at for cellular coverage in week 2. You are going to ‘deploy’ your network in this city. Try to find a reasonably grid-like section of streets in the city and measure the size of a city blocks for this city. For simplicity, we’ll just assume you are deploying on a rectangular grid the whole way. A rough estimate here is fine. As example, I would say that downtown San Diego city blocks are 90 m by 120 m.



Who’s [hopefully] providing you coverage, using what technology, in which city, how dense is your deployment, and what total area will it cover? Grab a screenshot of the city and overlay your device deployment area on top. (recall the fixed design point is 1,000 devices total; try to keep your deployment ~square; it need not actually map to the real streets, which probably doesn’t cleanly exist as a grid in most places ☺)

[1] Update: Jan 28, 2022 … Maybe not Sigfox, they just declared bankruptcy(!). Could still be interesting to do this analysis with them if you like, however.

Q1: Coverage

First, we are going to compare the coverage availability. What percentage of your devices would be covered by your cellular provider? What percentage would be covered by your non-cellular provider? If you had to make assumptions about coverage [especially for the non-cellular provider...], what assumptions did you make? Do you have enough information to assess how many devices are associated to each tower (or tower equivalent)?

Q2: Economic Cost

Next, we are going to compare the cost in dollars to deploy your system. Recall the analysis from week 2, for your new non-cellular provider, answer:

- What is the cost to provision 1,000 new devices (if any)?
- What is the cost for the data needs of this application?
    - Separately calculate the cost of the 1KB updates, the 10MB firmware pushes, and the total cost.
- What quality-of-service guarantees are available (if any)? [Note: `none` may be likely!]
    - Include things such as advertised speed/performance [even if not SLA contract promise]

Also copy over the estimates from your cellular provider from week 2, to have the comparison nearby.

Q3: Energy Cost

Then, repeat your energy analysis from last week, only now with a radio suited to for this network. Try to find datasheets or other sources for power draw that can separate the radio from the platform it is on (e.g. the STM Discovery board, a common prototyping board for LoRa, has a lot of extraneous components which result in significant power overhead — the energy analysis in this paper should be considered a pessimistic upper-bound, and your results should likely be better).

Be sure to consider key details such as expected data rate (and thus radio on time) of a deployed device. Will your device need to fragment packets? How much overhead will this add? For some examples of throughput estimation, this paper may be helpful.

- Energy during firmware-update event — This is primarily a downlink communication
- Energy during data-send event — This is primarily an uplink communication
- Energy during idle — What will cause your device to wake every 15m? [You may use an integrated, or separate timer]

Q4: Cost/Benefit of Engineering Alternatives?

That 10MB firmware update is *killer*. But we have come up with ideas in the past (e.g. Maté and its derivatives; delta-encoding app updates) that can drastically reduce update overhead. It costs around $360,000 USD annually to employ an embedded firmware engineer. Let's (generously…) assume it will take one engineer three months to design, implement, and test a revised firmware update approach that requires only 100 kB per update.

Which technology is the better choice *without* the revised firmware update process, why?

Would this revised firmware update change which technology is more cost-effective? If so, how long would it take for this development investment to pay off?