Wireless ad hoc Sensor Networks for City Street Light Maintenance

Lucas Addisi[1], Agustin Alexander[1], and Alexis Tcach[1 2]

[1] Instituto de Industria. Universidad Nacional de General Sarmiento
[2] Departamento de Computación. Universidad de Buenos Aires

Abstract. Wireless embedded devices, especially those of the open source hardware/software kind have seen rapid growth in recent years. This kind of devices when combined with ad hoc networks and sensor data can be a low-cost and highly scalable solution to many applications. In this paper we describe the experience of taking this concept of network infrastructure, developing application infrastructure and applying it, as a prototype, to a practical situation for a medium-sized city in the Buenos Aires province outer area, Argentina. We implement a novel sensor data delivery platform using open source hardware and software focused on streetlight maintenance at a fraction of the commercial implementation costs. To the extent of our knowledge this is the first system developed, deployed and documented in a Latin American country. Also presented in this paper is the system architecture as well as the design of hardware and software components as well as the results obtained.

Keywords: Ad hoc networks, MANET, WMSN, streetlights, environmental monitoring.

1 Introduction

Streetlight maintenance appears to be of major concern by City's administrators. Most cities have decentralized or uncontrolled streetlights systems which at most have a photocell sensor to switch them on at night. But there is mostly no other control over them. There exists some commercial products which depend on expensive *GSM* or *LTE* communication or proprietary *ad hoc* networks. [1, 2]

Mobile ad hoc networks (MANETs) are self-configuring, self-healing and dynamically adaptable networks of devices in which some of them become routers providing connectivity to the rest of them. There exist many different routing protocols used in this kind of networks. Optimized Link State Routing Protocol v2 [6, 8] or *OLSR2* for short is an *IETF* routing protocol for *MANETs*. It is categorized as a *Proactive* routing protocol, which seeks to maintain the routing

[0]E-mails:
`laddisi@campus.ungs.edu.ar` (L. Addisi),
`aalexander@campus.ungs.edu.ar` (A. Alexander),
`atcach@campus.ungs.edu.ar` (A. Tcach)

table of its nodes constantly updated so that at any given time each node knows which path to take in order to reach any other node in the network. Further analysis of ad hoc routing protocols and their classification can be found in Rosas et al. [7]

The main interest in our project are streetlights. Streetlight maintenance in cities can be troublesome for many reasons, first and foremost because their proper operation can only be verified at night. Secondly, because it requires either that it's citizens inform of malfunctioning lights or that the street light repair squad proactively searches for them. Furthermore, because of the many points of failure in them. The light post can have its energy cut of (due to vandalism or a blackout), the lamp can get burned out or the photocell can break. To make matters worse, some strange cases were reported where malfunctioning streetlights would suddenly start working again fooling the repairmen on arrival. As a side interest is the possibility to maintain an installed infrastructure for future exploitation, such as traffic light control and coordination, smart-grids interconnection, etc.

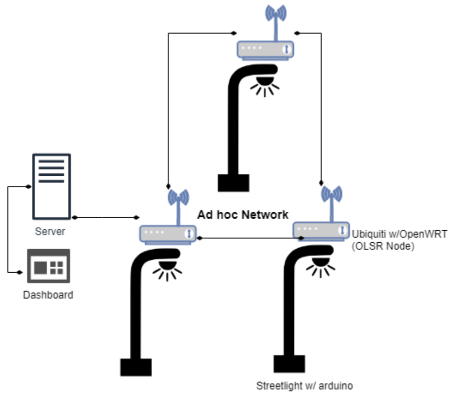
These solutions are extremely expensive for most Argentinean cities, not only to buy equipment but also the administrative software and mobile data subscription services. In fact, many cities around the world depend on the general public to report streetlight malfunctions[3] [4]. In order to cope with this kind of situations, a standardized city-controlled not vendor-tied system is needed that doesn't depend on expensive communication systems.

The city of San Miguel [9], which is a medium-sized city in Buenos Aires, decided to implement a wireless ad hoc sensor network system of interconnected streetlights and exit nodes in order to detect which ones are not operating properly. To do so current and temperature sensors are used. As a hardware constraint, the City of San Miguel already possessed many Ubiquity UniFi Outdoor Plus access point routers that were used by the city to share Internet in public areas. This routers had to be used as Wireless nodes.

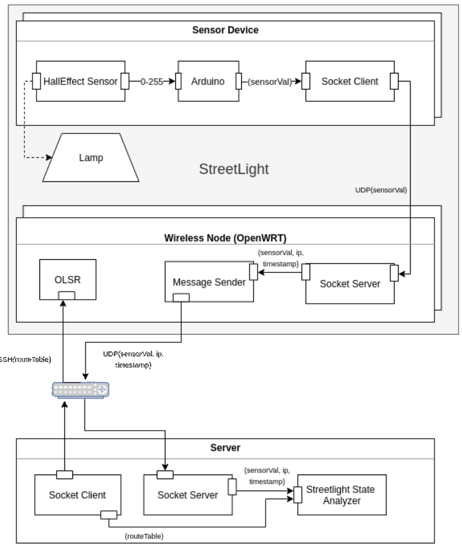
The system presented in this paper represents a step forward towards addressing these challenging issues using open-source hardware and software platforms, Arduino and OpenWrt. We not only address the architectural problem but also how to solve many of the aforementioned problems in a budget.

2 Overall System Architecture

The wireless ad hoc network's basic schema used for streetlights maintenance and control can be seen in 1a. Each streetlight is monitored and controlled by a sensor and has network a interconnection node which has the ability to connect in an ad hoc manner. Some of them connect to an access point which is part of the internal network of the city. This, in turn, is connected to a server explained in figure 1b



(a) Wireless ad hoc network basic schema for streetlights maintenance and control



(b) System architecture diagram

3 Design of Hardware and Software Components

In figure1b we can see that the sensor architecture.

The **sensor** consists of a Hall effect sensor *ACS712 30A* to gather current data from the streetlights and an Arduino UNO.[10], in charge of sampling data from the former and communicate with the wireless node.

The **wireless node** used is the 802.11b/g/n Ubiquity UniFi Outdoor Plus[12, 13]. This device's firmware was replaced by the Linux-based OpenWrt Operating System [11] allowing us to take control of the system and provide the user level interfaces that are needed. The Ubiquity UniFi, without removing its case, can only be connected to other devices and sensors using its LAN interface or via Wi-Fi. To do so a *W5100* arduino lan interface was used.

3.1 Network Design

The current sensor data is transmitted to the router every second from the Arduino using UDP and a fixed IP address in a separate subnet from the rest of the network, along with the Arduino's CPU temperature. The router runs a UDP server that listened for packets, adds a timestamp and retransmits them to a server via the ad hoc network.

One of the wireless nodes is connected using its remaining Ethernet port into an internal network provided by the city, making it the exit node in our system. This internal network has a path to the server where the data needs to be sent. The wireless nodes relay the sensor data inside UDP packets to each other using the OLSR protocol until they reach the exit node and finally the server.

A monitoring system to store and display node states that are derived from sensor and network data was also developed. This allows the streetlight repair squad to plan their trips.

The received data consist of current, temperature, IP and timestamp values. The first two are sent by the Arduino and the rest is added by the wireless node. This data, received in a JSON format (JavaScript Object Notation), is what allows the server to infer the node state. The nodes are displayed on a map in their approximate geographical locations. When data is received, the markers change their color according to the node's status.

In order to gather information on the node's state, internal *OLSR* information is used. As it is a proactive protocol, every node in the network knows the path and connectivity status. This allows us, without the need for any extra protocol or system, to know if a node is connected or not.

Information gathered, temperature, lamp's current and node's connectivity, the latter via *OLSR*, allowed us to determine

- The lamp is broken if the informed current is null.
- The photosensor is malfunctioning if there is current consumption on the light in the middle of the day and the other lights are off.
- The post's current is cut (a partial blackout) if many nodes are out of the *OLSR* network
- The Ubiquiti is broken or the post's current is cut (a partial blackout) if this only node is out of the *OLSR* network

4 Experimental Results

Results gathered were of two types; quantitative and qualitative. From the qualitative point of view, although the prototype test was too small to know the real impact of the system, final users were excited about the speed of information and its veracity. Another problem was the lack of metrics and information on present response times, as there is no information system nor exact report on which lights are malfunctioning.

During the test period, 4 circuits were broken. We found out that the original light configuration with some isolated electronics was not being used in actual streetlights deployments. This produced different results and electromagnetic noise that burnt the sensor circuit and hanged up the Arduino. This could be solved by using a Faraday cage around the Arduino and some custom made electronics to avoid the peak currents when lights were switched on or off.

Figure 2 shows a typical current data informed by 3 nodes. When the light is on, there is power consumption sensed. The aforementioned peaks can be seen when the light is switched on. We discovered that it was usual to find lights malfunctioning by seeing the consumed current by one of them, marked with a circle in the figure.

Fig. 2: Current over time of 3 streetlights.

5 Conclusions and lessons learnt

This hardware platform was chosen due to limitations by the city for this prototype. A production deployment would need to be embedded in one product. This sensor proved to be problematic due to the high current it was exposed to especially during lamp startup (Sodium streetlight lamps can have really high spikes in consumption during startup) and we had many of them burnt during our initial testing. To solve these issues a surge arrester (varistor) was added later on to the circuit.

As a future study we started to work on the possibility of using machine learning to automatically detect power anomalies and malfunctioning lights.

Bibliography

1. LightGrid Outdoor Wireless Control System, https://products.currentbyge.com/control-systems/lightgrid-outdoor-wireless-control-system
2. Decca Lighting, https://www.deccalighting.com/
3. Sacramento city, http://www.sacdot.com/Pages/StreetLightMaintenance.aspx
4. San Bernardino, http://www.specialdistricts.org/index.aspx?page=184
5. Westminster, https://www.westminsterlabour.org.uk/westminster-news/2019/03/15/westminster-council-lighting-contractor-goes-in-to-administration-leaving-half-the-street-lights-blazing-24-hours-a-day-but-council-keeps-councillors-in-the-dark-for-nearly-3-months/
6. T. H. Clausen, C. Dearlove, P. Jacquet, U. Herberg, The optimized link state routing protocol version 2, RFC 7181, RFC Editor (April 2014). https://tools.ietf.org/html/rfc7181
7. Rosas, Erika; Hidalgo, Nicolás; Gil Costa, Graciela Verónica; Bonacic, Carolina; Marin, Mauricio; et al.: Survey on Simulation for Mobile Ad-Hoc Communication for Disaster Scenarios; Springer; Journal of Computer Science and Technology 10.1007/s11390-016-1630-x https://ri.conicet.gov.ar/handle/11336/60810.
8. T. H. Clausen, P. Jacquet, Optimized Link State Routing Protocol (OLSR), Tech. rep., RFC Editor (Oct. 2003).
9. San Miguel (Buenos Aires), https://es.wikipedia.org/wiki/San_Miguel_(Buenos_Aires).
10. Arduino Uno, https://en.wikipedia.org/wiki/Arduino. Last accessed 24 April 2019
11. OpenWrt, https://en.wikipedia.org/wiki/OpenWrt.
12. Ubiquiti , https://www.ui.com/.
13. Ubiquiti Unifi Outdoor Plus, https://openwrt.org/toh/ubiquiti/unifi_outdoorplus.
14. EU202020, https://ec.europa.eu/clima/policies/strategies/2020_en#tab-0-0
15. Novak, T., Pollhammer, K., Zeilinger, H., & Schaat, S. (2014). Intelligent streetlight management in a smart city. Paper presented at the 19th IEEE International Conference on Emerging Technologies and Factory Automation, ETFA 2014, doi:10.1109/ETFA.2014.7005161