Optimization of Ultra-Narrowband Wireless Communication: an Experimental Case Study

Tomaž Šolc, Timotej Gale, Carolina Fortuna
Jožef Stefan Institute, Ljubljana, Slovenia
tomaz.solc@ijs.si, timotej.gale@ijs.si, carolina.fortuna@ijs.si

***Abstract*—Low-Power Wide Area Networks (LP-WANs) are emerging as a promising solution for connecting Internet of Things and Machine Type Communication devices. If ultra-narrowband (UNB) networks, a subclass of LP-WANs, reach predicted deployment numbers and densities, they will face two challenges: inter-technology and intra-technology interference. This paper proposes the first experimental architecture designed for the optimization of UNB networks. We illustrate its implementation on a case study of a SIGFOX network and the resulting extension of the existing LOG-a-TEC testbed. The proposed architecture enables context data collection, context model development, optimization and transmission control using rapid experimentation cycle approach enabled by flow based programming using Node-RED. Through preliminary results, we show the feasibility of PHY and MAC context data collection, point out challenges that are specific for UNB context modeling and discuss options for optimization. All datasets, context modeling and optimization tools used in the paper will be released as open source.**

***Keywords*—low-power wide-area network, ultra-narrowband, machine type communication, spectrum sensing, experimentation, optimization**

I. Introduction

Recently there has been growing interest in small autonomous interconnected sensors and actuators. Some predictions for the growth of the so-called Internet of Things now indicate trillions of deployed wireless devices by 2020 and densities of more than 1M devices per km^2 [1]. Existing widely deployed wireless technologies, such as LTE and IEEE 802.11 WLAN, have been optimized for low-latency, high-throughput links required by multimedia applications on laptop computers and smartphones. They are ill suited for machine type communication, where devices only need to infrequently transfer a few bytes of data at a time, with very relaxed requirements regarding latency and bitrate. In such cases, they exhibit low spectral efficiency and high overhead and do not scale well to high density of devices.

A number of technologies and standards have emerged recently that specifically address such use cases, the so-called Low Power Wide-Area Networks (LP-WAN) [2]. A subclass of these technologies employ ultra-narrowband (UNB) transmissions: SIGFOX [3], Weightless [4] and 3GPP Cooperative Ultra-Narrowband (C-UNB) [5]. These protocols use a physical layer that employs very low bit-rate transmissions (on the order of 100 to 1000 bits/s) using Binary Phase Shift Keying (BPSK) or Gaussian Frequency Shift Keying (GFSK) modulations with bandwidths on the order of 100 Hz to 1 kHz. This enables them to reach high spectral efficiency and low preamble overhead with short payloads [6].

Extremely narrow bandwidth allows for demodulation at low received signal strengths. Combined with use of sub-1 GHz bands with good propagation properties, UNB allows for large coverage areas with relatively low transmit powers, which makes it suitable for battery operated devices. Both SIGFOX and Weightless currently target the unlicensed 868 MHz Short-range devices (SRD) band for the European region. Extension to TV whitespaces (TVWS) as well as dedicated spectrum in the 694 - 790 MHz range [7] is possible in the future. In fact, Weightless was first developed for TVWS, but later moved to unlicensed bands.

If UNB networks reach predicted deployment numbers and densities, they will face two challenges:

1) Inter-technology interference comes from other technologies sharing the same frequency bands. Devices using independently developed technologies are most often not able to coordinate their use of the spectrum. Hence they have to rely on context information to avoid interference. Possible approaches are spectrum sensing, radio environment maps or some form of spectrum occupancy database. Inter-technology spectrum brokers could also be developed, enabling some form of inter-technology coordination [8], [9].

2) Intra-technology interference will result from increased device density in a network. This can lead to high collision rate in random access transmission schemes, requiring a large number of retransmissions. Intra-technology coordination mechanisms are easier to develop and deploy. For instance, the current approach in SIGFOX of choosing transmission time and channel pseudo-randomly can be optimized with a collision-avoidance scheme, possibly leading to a single transmission without significant impact on the overall performance.

These challenges are only partly addressed to date. Existing work on UNB optimization focuses on intra-technology interference and tends to be mostly theoretical in nature. In [10], the authors theoretically analyze the performance of a UNB network with a star topology and a random frequency selection based on the SIGFOX deployment. They consider random frequency selection both from a discrete set and a continuous range and discuss the effect of tolerances in the transmitter frequency reference. In [11], the effect of the number of frame repetitions on packet loss in an UNB network is analyzed. More broadly, it has been shown that in some cases collision avoidance allows for greater throughput in congested networks. For instance, [12] compares an ALOHA MAC protocol (randomized access without collision avoidance) with a carrier-sense multiple access (CSMA) in the context of an ad-hoc mesh network.

In this paper, we propose an architecture enabling the optimization of UNB networks and illustrate its implementation on a case study using optimization of a SIGFOX network. The architecture is generic and applicable to other case studies not considered in the paper, thus enabling more in depth studies of inter- and intra-technology interference in networks involving UNB. The proposed architecture enables context data collection, model development, optimization and transmission control using rapid experimentation cycle approach enabled by flow based programming using Node-RED. We also show preliminary results enabled by the proposed architecture. All datasets, the first of this kind, context modeling and optimization tools used in the paper are available as open source[1]. The resulting system extends the existing LOG-a-TEC testbed [13]. To the best of our knowledge, this is the first attempt to develop an experimental optimization system for UNB networks.

We structure the paper as follows. Section II describes the proposed architecture. Section III elaborates the considered case study while Section IV details the implementation and preliminary results. Finally, Section V summarizes the paper and outlines future work.

II. ARCHITECTURE

A generic architecture that enables the optimization of UNB wireless communication should contain four main blocks as illustrated in Figure 1 and discussed in this section. Each block covers a specific functionality needed for the optimization process.

The **context monitoring entity** is responsible for collecting context information that is relevant for supporting the optimization of the communication to minimize interference. This entity is relevant both for inter- and intra-technology optimization. Examples of context monitoring information are the number of detected collisions, signal-to-noise ratio of the received frames or packets, etc. This entity collects relevant information either by monitoring the physical environment or by connecting to other information providers such as base station or end-device reports. The design and implementation of such an entity for the inter-technology case is technically more challenging as it has to bridge between incompatible and often complementary technologies (i.e. short range vs. long range, high throughput vs. low throughput).

The **context modeling entity** uses context data collected by the context monitoring entity to model the operating environment. This block takes available context data and builds or updates relevant models for the system. For instance, using spectrum data and packet content, channel occupancy models or link quality estimates (LQE) can be modeled. This entity is more generic and conceptually less dependent on the actual technology target than the Context monitoring entity. However, the tuning of the actual models remains technology dependent. For instance, the development and tuning of an energy detector for UNB is different from an energy detector for wide-band transmissions.

The **optimization entity** consists of logic that consults the context models to decide on actions or configurations to be applied. For instance, the Optimization entity can retrieve channel

[1] https://github.com/ewine-project/sigfox-packet-datasets, https://github.com/ewine-project/sigfox-toolbox.

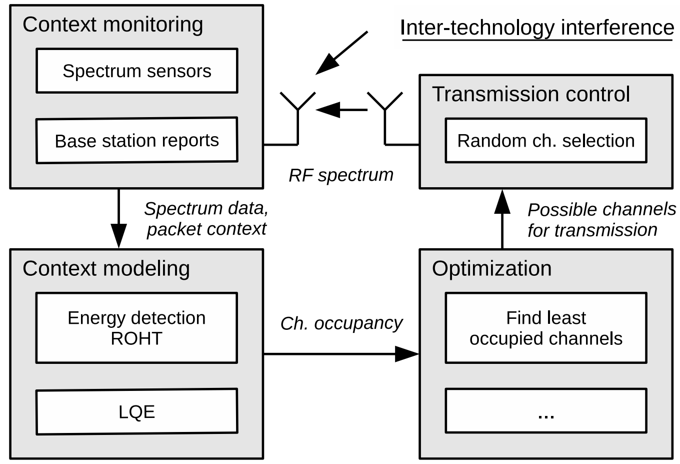

Fig. 1. Block diagram of the proposed architecture for optimization of UNB wireless communication with specific solutions adopted in our case study.

occupancy models from the Context modeling entity in the form of an occupancy table, and use these in the optimization process by selecting the first n least occupied channels. The entity can also perform pro-active optimization using predicted occupancy using for instance regression techniques rather than historic occupancy represented by occupancy tables. This entity implements mathematical optimization methods and tends to be the most generic from the four.

The **transmission control entity** configures the transmitters under optimization with the values of the parameters chosen by the Optimization entity. For instance, it configures the channel numbers used by the transmitter. This entity interacts directly with the technology under optimization, thus its implementation is technology specific. Its complexity increases for inter-technology optimization use cases.

III. CASE STUDY

As a case study, we consider the UNB SIGFOX LP-WAN in the unlicensed 868 MHz band. It provides low-throughput, long-range, energy-efficient connectivity to fixed-location devices like wireless sensor nodes and objects in the so-called Internet of Things. Radio links in the SIGFOX network are exclusively between a device and a base station. There is no direct device-to-device communication. The network supports both downlink (base station-to-device) and uplink (device-to-base station). However, most use cases require only uplink (for example, a sensor periodically reporting measurements) and hence support for downlink communication is optional for devices in the network.

The SIGFOX network uses differential binary phase shift keying (DBPSK) modulation for the uplink communication. The physical bit rate is 100 bits per second with packet length of approximately 300 bits. The uplink band at 868.130 MHz is divided into 1500 microchannels in a 100 Hz raster. Devices employ a proprietary frequency hopping and frame repetition pattern in which each frame is transmitted three times on three pseudo-randomly chosen microchannels.

The aim of our optimization is to study whether the number of retransmissions from devices can be decreased without decreasing the overall performance of the network. This would

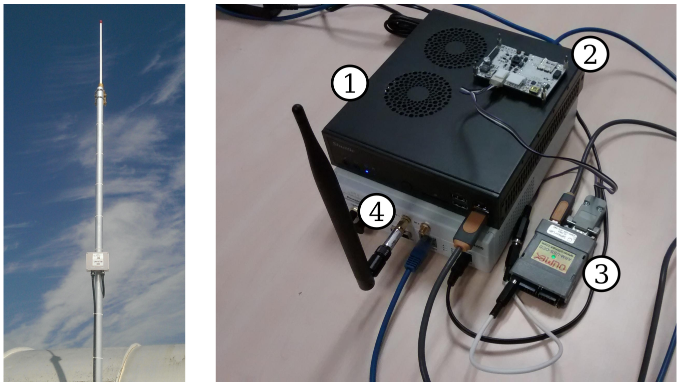

Fig. 2. Left: base station antenna. Right: SIGFOX test device - (1) miniature PC, (2) embedded board with modem firmware, (3) serial interface between PC and embedded board, (4) USRP N200 transceiver.

possibly come at the cost of having to transmit some context data from the base station back to devices. Our rationale is that spectrum occupied by frame retransmissions scales with the number of devices, while context data transmission can be done once for all devices. Hence this approach should be more efficient in a dense SIGFOX network.

IV. IMPLEMENTATION

The implementation of our system required 1) SIGFOX devices, acting as transmitters, 2) SIGFOX base station, acting as receiver and 3) the four additional entities described in Section II. The SIGFOX base station (Figure 2, left) allowed for significant deviation from the standard SIGFOX protocol and proved to be usable for experimentation without any further modifications. Implementation of other components is described in the following sections:

A. SIGFOX device

A typical SIGFOX device consists of an application processor and a sealed RF module (modem). Modems with production firmware do not allow control over low-level details such as time and channel of frame transmissions and repetition pattern. Modification of existing modem firmware was considered too time consuming. Hence we opted to create a new implementation of the SIGFOX protocol in a device that would enable rapid development and testing of optimized protocols.

Our test device was based around a miniature, Intel-based PC running a GNU/Linux operating system. We implemented the physical layer in Python using the GNU Radio framework [14] and NumPy [15]. We used the Ettus USRP N200 transceiver [16] with the SBX daughterboard as the radio front-end. To avoid re-implementing the higher-level details of the SIGFOX protocol, we re-used a binary firmware library that is provided by SIGFOX to modem manufacturers. This library was linked with a mock application and uploaded to a compatible embedded device. The firmware called library functions on request from our software on the PC and intercepted calls that would ordinarily go to the embedded transceiver driver. This allowed us to use the unmodified library to perform SIGFOX packet header handling and cryptographic functions while leaving the physical layer details to our own software on the PC. We only implemented the uplink functionality. A photograph of the device is shown in Figure 2, right.

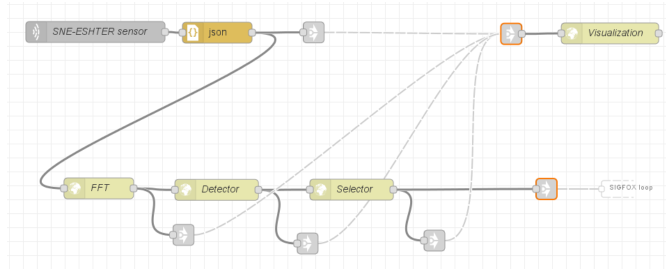

Fig. 3. Node-RED flow that receives spectrum data from a sensor, performs energy detection, calculates channel occupancy and performs visualization.

The rest of our setup similarly followed the requirement for a rapid experimentation cycle. We chose to base high-level functionality on Node-RED, which is a browser-based, visual editor for flow-based programming [17]. Where Node-RED flows interacted with external components (such as spectrum sensors, the test device and the SIGFOX base station), the connection was made using HTTP APIs. Some parts of the system required high data I/O rates or computationally intensive tasks (for instance, tasks handling raw data from spectrum sensors). These were implemented as separate Python processes and again connected with Node-RED over HTTP. An example Node-RED flow is shown in Figure 3.

B. Context monitoring

The Context monitoring entity collected MAC and PHY level context data as follows.

Firstly, we recorded data related to each packet transmitted in the network. An example packet record in JSON format is shown in Figure 4. The test device recorded the transmission time based on the device clock, central frequency in Hz for each frame, amplifier gain or attenuation in dB and number of frame repetitions. The SIGFOX base station provided measurements related to the physical layer on the receive end: receive time based on base station clock, received signal strength indicator (RSSI) in dBm and signal-to-noise ratio (SNR) in dB (both current packet SNR and average over last 25 packets). It also recorded base station and device identifiers. For packets that were successfully decoded by the base station, these measurements were recorded alongside the received packet payload. The context data from the receive side was matched with the transmitter settings based on a sequence number contained in the packet payload. This matching process yielded another source of context: which packets were received and which were lost.

Secondly, we used spectrum sensors to provide periodic samplings of the SIGFOX uplink band. We used two SNE-ESHTER spectrum sensors, one co-located with the transmitting device and one co-located with the base station. SNE-ESHTER is a low-cost, custom designed VHF/UHF receiver based on an NXP TDA18219 integrated silicon tuner and a ARM Cortex M3 microcontroller. It is capable of baseband signal capture of up to 25000 samples with rates up to 2 Msamples/s. The sensor has been designed for out-door

```
{ "rx": {
    "avgSnr": 10.67,
    "snr": 8.73,
    "time": 1477563622,
    "rssi": -109.00,
    "station": "0BF2",
    "device": "1CF14C"
  },
  "tx": {
    "attenuator": 0.0,
    "pga_gain": 0.0,
    "time": 1477564048.7,
    "repetitions": 1,
    "frequency": [
      868194200
    ] } }
```


Fig. 4. Example of context data recorded for a single packet transmitted in the SIGFOX network. The packet was successfully received by the base station since the receive-end data (the `rx` property) is present.

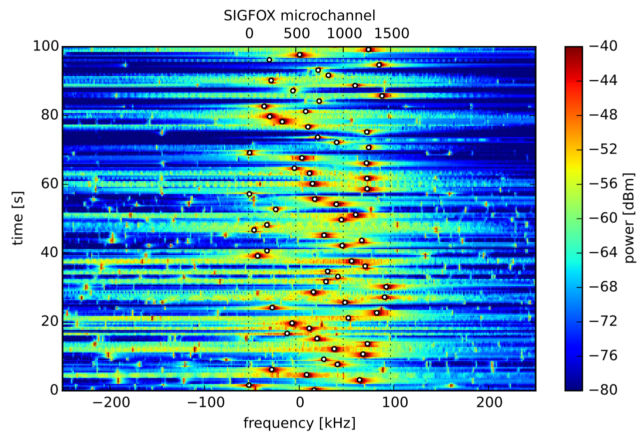

Fig. 5. Visualization of the spectrum sensing data from the sensor co-located with the transmitting device. Circular markers mark times and frequencies of SIGFOX frame transmissions (and associated packet context data).

deployments in testbeds and can be remotely controlled over an Ethernet connection [18].

Both spectrum sensors were set to record 5000 samples at 1 Msample/s with a central frequency of 868.2 MHz. Hence they recorded activity in all SIGFOX microchannels, together with some margin above and below. Data recorded from the sensor included sensor identifier, time of each record based on sensor clock and vectors of signal samples. Compared to packet context, spectrum sensing produced a much larger amount of data. Hence, we used a binary format for storage. An excerpt from the recorded spectrum data is visualized in Figure 5.

Since multiple clocks and frequency references were used in the system, measurements from all sources needed to be translated onto a common reference in order to present a consistent picture. The offset between device and base station clocks was determined by matching individual packet records through sequence numbers. We ignored the time-of-flight delay. Offset between clocks and frequency references in the test device and spectrum sensors were determined by matching transmission times and frequencies with the corresponding peaks in the power spectral density.

C. Context modeling

The Context modeling entity developed channel occupancy models using MAC and PHY context.

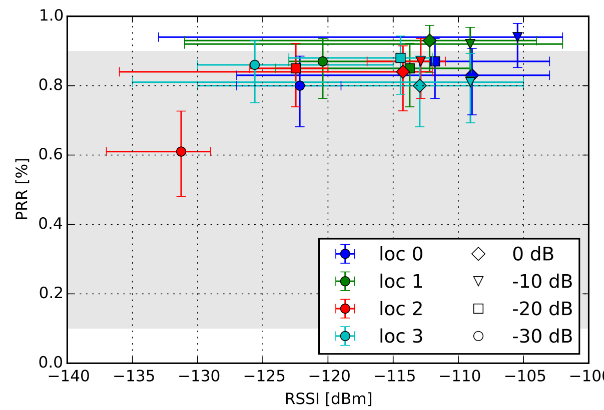

Fig. 6. PRR vs. RSSI for four locations and four transmit gains. Shaded region shows PRR range for intermediate links, with good links above and bad links below. Error bars show 99% confidence interval (N=100).

First, we used a machine learning approach to predict which channels are likely to be of better quality for the next n packets. For this we defined a classification task aiming to automatically predict which links will behave as "good links" with packet received ratio (PRR) above 90%, "intermediate links" with $10\% < \text{PRR} < 90\%$ and "bad links" $\text{PRR} < 10\%$. For validating the proactive LQE model, received packet context data (see Figure 4) was collected for 16 links using four different physical locations and four different transmit powers. Transmit power was varied by changing the USRP programmable gain amplifier (PGA) gain in steps of 10 dB. With a fixed external 30 dB attenuator this resulted in total transmit gains between -30 dB and 0 dB. The transmitting device sent 100 packets in total for every distinct location and gain configuration, totaling 1600 measurements.

The distribution of the data regarding PRR and RSSI is presented in Figure 6. Using the PRR classification above, most of the measured links fall within the intermediate class. There was a significant variation in RSSI reported by the base station. For 7 links, the RSSI varied by more than 20 dB. In total, the campaign enabled the collection of 37% good links and 63% intermediate links.

We encountered difficulties collecting data with low PRR, hence our dataset does not currently cover bad links. The SIGFOX base station verifies the received packets against an internal sequence counter. If a large discrepancy accumulates between the internal counter and the sequence numbers in received packets the base station will drop all future received packets with a "trash sequence number" error. This frequently occurred in our case when performing measurements with low PRR and introduced extra packet loss into our measurements that was not due to link quality. The sequence counter reset required manual intervention and was at that time impossible to integrate into our automated measurement procedure.

We considered a decision tree model (J48 information theoretic algorithm) that is quick to train, tends to have moderate to good performance and, most importantly, it also gives information on which input data (feature) is most informative and more relevant. This is important for understanding the data and engineering classification and regression models using

TABLE I. LQE CLASSIFICATION FOR THE BEST PERFORMING FEATURE VECTORS.

Feature vector	Correct	Incorrect
rssi, avg(rssi), std(rssi), avg(snr), std(snr), avgSnr	78.82%	21.17 %
avg(rssi), std(rssi), std(snr), avgSnr	78.01 %	21.98%
avg(rssi), avg(snr), std(snr), avgSnr	77.42%	22.57 %

more advanced algorithms [19].

As reported in Table I, the model was able to accurately classify up to 78% of the links. All of models use the average RSSI, when available, in the root of the tree, meaning that, in the available dataset, this was the most reliable predictor for the link quality. The second most important predictor was the standard deviation of the RSSI while the instant RSSI only came third.

Challenge The usefulness and accuracy of link quality estimation is a debated topic in the research community [20], [21]. Our working hypothesis is that link quality estimation is a useful abstraction and is able guide the channel selection and the number of message repetitions.

Second, we established the channel occupancy using spectrum sensing. SIGFOX transmissions can be decoded by the base station at very low signal powers due to the extremely narrow bandwidth which limits the in-band noise power. Hence we require a spectrum sensor that is also able to detect similarly weak interfering signals. At the same time, the sensor must also be able to provide sufficient frequency resolution to be able to determine occupancy on the level of individual microchannels.

It is well known that energy detectors are not able to reliably detect weak transmissions due to the SNR wall [22]. This led us to apply covariance-based detection to this problem. In our previous work we found that such detectors have good performance in detecting the presence of narrowband signals [23]. However, a covariance-based detector by itself does not provide fine-grained frequency information on the level of SIGFOX microchannels. Applying detection to signals filtered to individual microchannels appeared impractical due to excessively long time windows required. Hence covariance detection currently seems most useful in 1) detecting interferers covering many microchannels and 2) in avoiding interference based on time, and not frequency, of transmission.

In light of these limitations, we decided to use energy detection based on estimated power spectral density and a dynamically adjusted threshold. Using this approach the theoretical frequency resolution is inverse of the time window length used for the estimation of power spectral density. In this case required time windows were easily achievable using our sensing hardware. Despite the fact that such an energy detector will likely misclasify many microchannels as vacant, we felt that its increased frequency resolution will provide more optimization options.

We estimated the power spectral density in the SIGFOX uplink band using a fast Fourier transform (FFT) of the

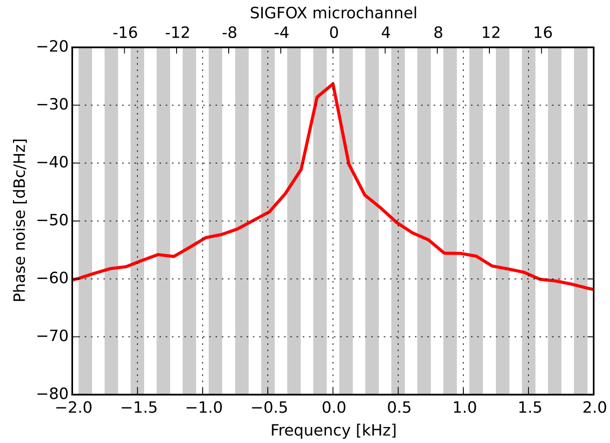

Fig. 7. Comparing phase noise of the spectrum sensor to the bandwidth of SIGFOX microchannels.

spectrum sensor signal samples. The estimate had a resolution of 200 Hz, hence each FFT bin covered two 100 Hz SIGFOX microchannels. Our detector performed a binary decision for each FFT bin based on a threshold value. The threshold was determined separately for each 10 consecutive bins. If the power spectral density for a bin was above the threshold, the two microchannels covered by that bin were considered occupied. In the other case, they were considered vacant.

To determine the threshold, a Recursive One-sided Hypothesis Testing (ROHT) algorithm [24] was used, based on the power spectral density in the log scale. For each group of 10 FFT bins, a separate instance of the algorithm was performed. We used a history of 1000 power spectral density estimates, 90% confidence interval, $\varepsilon = 0.5$. The threshold was recalculated every 20 samples. The sensor provided one record, and hence one power spectral density estimate, per approximately 500 ms. This means that the history length was approximately 8 minutes and recalculation interval approximately 10 seconds.

Challenge Our spectrum sensor has a relatively high phase noise (see Figure 7). This limits the accuracy of the channel occupancy table - a transmission that occupies only one microchannel will appear to our detector to occupy several neighboring microchannels. On the other hand, it is questionable how valuable a precise occupancy table is in practice. Most SIGFOX devices themselves do not contain precision oscillators [10] (e.g. with an ordinary 10 ppm quartz resonator, transmitter has an offset of up to 100 microchannels).

D. Optimization entity

The Optimization entity uses 1) predicted LQE, 2) historical channel occupancy statistics or 3) a mix of the two to determine the most suitable transmit channels for the SIGFOX devices. The predicted LQE in the form of PRR value (i.e. using regression learning) will be used to perform adaptive retransmissions. For instance, for a "good link", only one transmission is needed, while for other two classifications, more retransmissions will be needed. The qualitatively predicted LQE (i.e. using classification) will be used to select better links and minimize retransmissions. Complemented by channel occupancy statistics, different set-ups using these techniques will be used to optimize UNB network represented by SIGFOX. For evaluating the performance of each optimization strategy, the default SIGFOX transmission mechanisms will be ran as a baseline, against which the optimized channel selection will be compared.

The best performing optimizations will be selected and considered for implementation in terms of complexity and messaging overhead. One option would be to have the optimization system implemented in each device while the opposite option is to have the base station periodically transmit configuration data to the devices.

The currently available implementation uses a simple optimizer based on historical occupancy statistics for microchannels that was calculated over a time window of approximately 8 minutes (1000 spectrum sensor reports). Each device receives a list of 20 least occupied microchannels from which they randomly select a candidate for the transmission.

E. Transmission control entity

The device sets the transmitter parameters and performs transmissions according to the microchannels and required number of retransmissions provided by the optimizer.

V. SUMMARY AND FUTURE WORK

In this paper, we proposed a generic architecture that enables experimental optimization of UNB communication. The architecture can be implemented and integrated in existing experimental facilities allowing users (researchers, technology experts) to collect context data, model and optimize the network and experimentally verify the results. The validation of the proposed architecture is performed on the SIGFOX UNB network. We used a standard base station, developed a flexible end-device and implemented the blocks of the proposed architecture. Our preliminary results show the feasibility of PHY and MAC context data collection, channel occupancy, link quality modeling and network optimization. The UNB context data and other tools developed in this effort will be openly released. Furthermore, the tools and set-up extend the existing LOG-a-TEC testbed with UNB functionality.

In the future we plan to 1) run a larger context collection campaign that will also cover bad links, using new functionality provided by SIGFOX to programmatically reset base station sequence counter, 2) develop more accurate LQE classifiers and regression models, 3) finish the implementation of the optimizer and 4) using the developed system, perform a complete study on UNB technology optimization in presence of inter- and intra-technology interference.

ACKNOWLEDGMENT

We are thanking SIGFOX for providing the base station, modem firmware library and support for the base station cloud API. This work was partly funded by the EC under eWINE (688116) and Slovenian Research Agency (P2-0016).

REFERENCES

[1] The 5G Infrastructure Public Private Partnership (5GPPP), "5G empowering vertical industries," 2016. [Online]. Available: https://5g-ppp.eu/wp-content/uploads/2016/02/BROCHURE_5PPP_BAT2_PL.pdf

[2] Nolan, K. E. and Guibene, W. and Kelly, M. Y., "An evaluation of low power wide area network technologies for the Internet of Things," in International Wireless Comm. and Mobile Computing Conference (IWCMC). IEEE, 2016, pp. 439–444.

[3] SIGFOX, "Makers & Developers portal," 2016. [Online]. Available: http://makers.sigfox.com

[4] Weightless SIG, "Weightless-N," 2016. [Online]. Available: http://www.weightless.org/about/weightlessn

[5] 3GPP, "C-UNB technology for Cellular IoT - Physical Layer (GP-150057)," Shanghai, China, 2015.

[6] Lassen, T., "Long-range RF communication," Texas Instruments, 2014.

[7] European Commission, "Commission proposes to boost mobile internet services with high-quality radio frequencies," 2016. [Online]. Available: http://europa.eu/rapid/press-release_IP-16-207_en.htm

[8] A. M. Voicu, L. Simi, and M. Petrova, "Inter-Technology Coexistence in a Spectrum Commons: A Case Study of Wi-Fi and LTE in the 5-GHz Unlicensed Band," IEEE Journal on Selected Areas in Communications, vol. 34, no. 11, pp. 3062–3077, Nov 2016.

[9] R. Ferrus, O. Sallent, and R. Agusti, "Interworking in heterogeneous wireless networks: Comprehensive framework and future trends," IEEE Wireless Communications, vol. 17, no. 2, pp. 22–31, 2010.

[10] Do, Minh-Tien and Goursaud, Claire and Gorce, Jean-Marie, "On the benefits of random FDMA schemes in ultra narrow band networks," in 12th Int. Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt). IEEE, 2014, pp. 672–677.

[11] Mo, Y. and Do, M.-T. and Goursaud, C. and Gorce, J.M., "Optimization of the predefined number of replications in a Ultra Narrow Band based IoT network," in Wireless Days (WD). IEEE, 2016, pp. 1–6.

[12] Ferrari, G. and Tonguz, O. K., "MAC protocols and transport capacity in ad hoc wireless networks: Aloha versus PR-CSMA," in Military Communications Conference (MILCOM), vol. 2. IEEE, 2003, pp. 1311–1318.

[13] T. Šolc, C. Fortuna, and M. Mohorcic, "Low-cost testbed development and its applications in cognitive radio prototyping," in Cognitive Radio and Networking for Heterogeneous Wireless Networks, ed. A. F. Cattoni, L. De Nardis, J. Fiorina, F. Bader, M. G. Di Benedetto. Springer, 2014, pp. 1–45.

[14] GNU Radio Foundation, Inc., "GNU Radio, the free & open software radio ecosystem." [Online]. Available: https://gnuradio.org

[15] "Numpy," 2016. [Online]. Available: http://www.numpy.org/

[16] Ettus Research, "USRP N200/N210 Networked Series," 2012. [Online]. Available: https://www.ettus.com/content/files/07495_Ettus_N200-210_DS_Flyer_HR_1.pdf

[17] IBM, "Node-RED," 2016. [Online]. Available: http://nodered.org

[18] T. Šolc, "Adopting WINNF Transceiver Facility for Spectrum Sensors," in Wireless Innovation Forum European Conf. on Comm. Technologies and Software Defined Radio (WInnComm-Europe). Erlangen, Germany: Wireless Innovation Forum, 2015.

[19] M. Kulin, C. Fortuna, E. De Poorter, D. Deschrijver, and I. Moerman, "Data-driven design of intelligent wireless networks: An overview and tutorial," Sensors, vol. 16, no. 6, 2016.

[20] N. Baccour, A. Koubâa, L. Mottola, M. A. Zúñiga, H. Youssef, C. A. Boano, and M. Alves, "Radio link quality estimation in wireless sensor networks: a survey," ACM Transactions on Sensor Networks (TOSN), vol. 8, no. 4, p. 34, 2012.

[21] D. Gokhale, S. Sen, K. Chebrolu, and B. Raman, "On the feasibility of the link abstraction in (rural) mesh networks," in INFOCOM 2008. The 27th Conference on Computer Communications. IEEE. IEEE, 2008.

[22] S. Anant, N. Hoven, and R. Tandra, "Some fundamental limits on cognitive radio," in Allerton Conference on Communication, Control, and Computing, 2004, pp. 1662–1671.

[23] T. Šolc, "Low-cost spectrum sensor for ultra-narrowband transmissions," in 8th Jožef Stefan Int. Postgraduate School Students' Conference (IPSSC). Ljubljana, Slovenia: Jožef Stefan Int. Postgraduate School, 2016.

[24] Datla, D. and Wyglinski, A. M. and Minden, G. J., "A Spectrum Surveying Framework for Dynamic Spectrum Access Networks," IEEE Transactions on Vehicular Technology, vol. 58, no. 8, pp. 4158–4168, 2009.