Game Theoretic and Auction-based Algorithms towards Opportunistic Edge-Processing in LPWA LoRa Networks

Mo Haghighi1, Zhijin Qin2, Davide Carboni1, Usman Adeel1, Fergusn Shi2, Julie A. McCann2
1Intel Labs, 2Imperial College London
mo.haghighi@intel.com

Abstract—Low Power Wide Area (LPWA) networks have been the enabling technology for large-scale sensor and actuator networks. Low cost, energy-efficiency and longevity of such networks make them perfect candidates for smart city applications. LoRa is a new LPWA standard based on spread spectrum technology, which is suitable for nodes with long battery life, low data rate, and bi-directional communication. In this paper we will demonstrate a use-case inspired model in which, end-nodes with multiple radio transceivers (LoRa/WiFi/BLE) may facilitate edge processing and interconnect multiple networks. The model primarily focuses on switching radio communications opportunistically and adaptively based on the application requirements and dynamic radio parameters.

Keywords—Game theory; IoT; WSN; LPWA; LoRa; ICRI; Edge; Smart City; Auction; Utility;

I. INTRODUCTION

Wireless sensor networks (WSNs) have become a major technology for a wide variety of applications ranging from medical, to military and environmental monitoring. Conventional applications often involved a limited number of sensors spread across an environment of interest, in order to aggregate various parameters in a centralised manner.

Modern applications however, require data to be aggregated and processed in the field and in a distributed fashion, in order to derive context-aware results inspired by all network entities. Attaining context-aware operation requires collaborative processing of captured data using wide variety of computational functions integrated in the end-nodes, which ultimately leads to network-wide intelligence. Integrated intelligence facilitates autonomous decision-makings using global and local knowledge of network entities. Such intelligent decision-making becomes crucial when optimising resources and network operations in order to tackle constraints of WSNs. Dynamic reorganisation and runtime reprogramming of network entities are examples of such operations required by diverse applications.

From the applications perspective, integrated intelligence is a significant key aspect of many applications involving autonomous actuations, which are prevalent in the IoT domain. Those applications often involve several actuation operations, which may result from the aggregation of multiple sensing operations on their given environmental conditions. The computational process involved in such applications includes sophisticated sets of logical and mathematical functionalities, often implemented in a collaborative fashion amongst a large number of sensors and actuator.

Such collaborative functionalities, also known as edge-processing, necessitate a relatively powerful MCU/MPU and sufficient memory for processing raw data and storing the results. Therefore, nodes need to be equipped with more powerful components compared to the conventional WSNs. In addition to the need for more powerful resources, such advanced functionalities are power-hungry processes, which take considerable amount of energy in order to handle the high footprints on the memory and processor. However, most energy expenditures are towards wireless communication between sensor nodes.

Low-power wide-area network (LPWAN), enables the City of a Billion Devices to communicate by contributing to city-wide network capacity optimisation. In LPWANs, long range transmission with low energy consumption is critical to enable the promise of ten year system lifetime with minimum maintenance. In heterogeneous wireless networks with multiple radio technologies (such as LoRa, BLE, 802.15.4 and WiFi etc.), an important question that arises is how should a node select the best radio; and within a given radio range, what are the most suitable parameters to be used at any given time, environmental conditions and most importantly the varying application demands?

Radio selection has been extensively studied in heterogeneous networks [8-10], particularly in cases when there is assistance from the network, or when a central controller is able to distribute users across networks in order to optimize some notion of system performance. However, most of the existing work focuses on the radio selection among cellular network and some other network such as Wi-Fi, which are supposed to be conducted at mobile phones. No particular efforts have been put on the LPWAN, which is believed to the solution to provide massive connections in IoT. Additionally, the data features for IoT scenarios are quite different from mobile phones. Sensors in LPWANs are more critical to energy consumption than mobile phones. In order to solve the above concerns, the adaptive communication algorithm is more than necessary to enable the sensors switching radio opportunistically to minimize the energy consumption at each node but with guaranteed qualities of service per node and over the entire network. For a LPWAN within city-scale, a distributed adaptive radio selection is required to make sure the algorithm is easily scalable. The sustainability is also need to be guaranteed for the considered network.

In the distributed radio selection algorithm, we focus on an agent-centric approach, in which each agent will manage a resource pool. Based on the current network situation, the agent will make decisions to select the appropriate radio for the nodes covered by itself. No prior information about nodes managed by other agencies need to be known. This can be achieved by using game theory, in which the agent will only strive to make the best strategy to minimize energy consumption at node level and maximize the throughput locally regardless of the nodes managed by other agents. Meanwhile, the agent also needs to determine the optimal parameters for the selected radio. Taking LoRa radio as an example, the parameters such as spreading factor, channel frequency, bandwidth, and transmission power need to be
determined, in order to maximize the network throughput and minimize energy consumption at each sensor node and across the network city-wide. By using such an approach, the energy consumption at sensor nodes can be minimized as the strategy process is moved from sensor nodes to its agents.

In this paper we will introduce a model that uses:

1. **Game Theory** as a tool for non-collaborative analysis by the end-nodes in order to maximise their utilities, including the selection of radio parameters, regardless of the actions of other network peers.

2. **Auction-based algorithms** to create a collaborative process for negotiating resources utilisation, in particular the radio-head selection.

The aforementioned operations can only be implemented in a device that provides the essential underlying resources and services, most notably multiple radio-heads and software services required for running the algorithms. For this purpose we will LoRaBox instrument, which will be explained in the next section.

Figure 1 shows multiple radio networks in Queen Elizabeth Olympic Park (QEOP) in London. Intel Collaborative Research Institute (ICRI) will be utilizing QEOP living lab as a shared common research infrastructure to deliver various research agenda surrounding billion device challenge for future IoT networks, including adaptive opportunistic communication.

LoRaBox
LoRaBox is a multi-radio testing instrument designed for sending and receiving data packets via:

- **LPWA LoRa**
- **Bluetooth Low Energy (BLE)**
- **Wi-Fi**

LoRaBox is a portable embedded system based on Intel Edison compute module, running Yocto Linux and Java SE. LoRaBox integrates a LoRa transceiver (Microchip RN2483), 2x16 character LCD, rotary encoder switch, a solar panel and a 4000 mAh battery. LoRaBox runs an agent-based program developed in Java SE.

LoRaBox captures data via one radio transceiver, processes the data, and forwards it over the same radio or other transceiver(s). A simple interconnectivity functionality of the LoRaBox is capturing BLE beacons and transmitting the data over LoRa (ISM 868 MHz) and WiFi. Figure 2 shows a BLE microclimate sensor based on Intel Quark SoC (Curie) on the left and the LoRaBox on the right.

The algorithms integrated in the device allows the instrument:

1. to switch between different radios opportunistically depending on the signal strength and energy requirement
2. dynamically switch the radio parameters in order to optimise the connection quality and data rate.

Dynamic parameters that we mentioned earlier are mainly targeting LPWA LoRa in 2 major categories of Radio and Energy.

**Radio:**

- (SF) Spreading Factor
- (CR) Coding Rate
- (CH) Channel
- (BW) Bandwidth

**Energy:**

- (Tx) Transmission Power
The algorithms explained in the next section are based on auction and game-theory methodologies as described in [1-7]. For the purpose of this experiment, our proposed algorithms aim to dynamically adjust the above-mentioned parameters in order to meet the application requirements with respect to the desired:

- Latency
- Data Rate
- Energy Consumption

Finally, our algorithms tune their performance by the taking into account the following near real-time measurements received into the LoRaBox from the main gateway via Wi-Fi and our cloud platform:

- RSSI
- SNR
- Airtime
- Packet Loss

In order to clarify the aforementioned process, Figure 3 illustrates the sequence in which LoRaBox and Gateway interact via EnableIoT cloud platform, including their data transmission.

As figure 3 shows, data packets are initially transmitted to the LoRa gateway on the Orbit. LoRa gateway extracts the packet’s properties including the RSSI, SNR, Channel number, Frequency and its spreading factor. All that data is transferred over a web socket to Intel EnableIoT cloud, which is an IoT analytic dashboard and cloud storage. In the third phase, data are pushed back to LoRaBox via the park public Wi-Fi, which covers the majority of the locations in the park. LoRaBox identifies and analyses the received properties of the packets using their unique ID numbers.

It then applies the algorithms to adjust the transmission power, spreading factors and etc. There also exists a web UI, by which live data packets and the locations they are sent from are displayed over the map of the park. Figure 4 shows, as packets arrive, the UI shows the radio parameters of the packet as well as the data in both encrypted and Hex forms. It is worth noting that all LoRa communications between the LoRaBox and LoRa Gateway are taking place using LoRa point-to-point mode without acknowledgement, and not LoRaWAN.

<table>
<thead>
<tr>
<th>Mode</th>
<th>BW (KHz)</th>
<th>CR</th>
<th>SF</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>125</td>
<td>4/5</td>
<td>12</td>
</tr>
<tr>
<td>2</td>
<td>250</td>
<td>4/5</td>
<td>12</td>
</tr>
<tr>
<td>3</td>
<td>125</td>
<td>4/5</td>
<td>10</td>
</tr>
<tr>
<td>4</td>
<td>500</td>
<td>4/5</td>
<td>12</td>
</tr>
<tr>
<td>5</td>
<td>250</td>
<td>4/5</td>
<td>10</td>
</tr>
<tr>
<td>6</td>
<td>500</td>
<td>4/5</td>
<td>11</td>
</tr>
<tr>
<td>7</td>
<td>250</td>
<td>4/5</td>
<td>9</td>
</tr>
<tr>
<td>8</td>
<td>500</td>
<td>4/5</td>
<td>9</td>
</tr>
<tr>
<td>9</td>
<td>500</td>
<td>4/5</td>
<td>8</td>
</tr>
<tr>
<td>10</td>
<td>500</td>
<td>4/5</td>
<td>7</td>
</tr>
</tbody>
</table>

Spreading Factor (SF), which is the modulation type for spread spectrum has a major impact on airtime, receiver sensitivity and bandwidth. The higher the SF, the higher the receiver sensitivity, whereas SF is inversely proportional to bandwidth and increasing the SF will increase the airtime. For the transmission power Tx, we have 5 options:

<table>
<thead>
<tr>
<th>Tx (dbm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>7</td>
</tr>
<tr>
<td>10</td>
</tr>
<tr>
<td>14</td>
</tr>
</tbody>
</table>

For the purpose of experiments described in this paper, LoRa gateway is fixed at a single frequency band and all transmissions are done using Channel 14 at 868.0 MHz. Therefore our proposed algorithms will not modify the channel number and frequency.
II. METHODOLOGY

In this section our game theoretic model and auction-based techniques will be described briefly before we detail our methodology in the next section.

Game Theory and Auctions

The proposed game theoretic method is based on our previous work [3], in which data trustworthiness of sensor networks was investigated using two models:

1. Intrusion Detection Model (IDM)
2. Intrusion Prevention Model (IPM)

The aforementioned models were used not only to avoid faulty and malicious data from the compromised nodes, but also to recover them.

IDM presents a game between a defender who supervises a selected number of nodes (without the knowledge of which ones are going to be attacked), and an attacker who attempts to compromise some nodes randomly in order to generate as much false sensory data as possible. The first important factor is the number of nodes being supervised by the defender in a region, which is a vital aspect of defender’s strategy. The second factor is every node’s coefficient of significance, which represents the level of trust for the data it transmits.

Figure 5: Schematic description of IDM

Figure 5 shows the schematic description of IDM between the attacker and defender, in which Tolerance represents the minimum portion of total data that is considered untrusted, having known the portion of the trusted and total data.

Attacker’s Payoff Matrix in figure 5 is a matrix populated using Attacker’s Payoff function as below:

\[
AP = \left( \frac{\text{iS}}{\text{Fs}} \geq t \right) \times rcn + s \times cps - a \times cpa + t \times tc
\]

(1)

AP = Attacker’s Payoff, is = incorrect sum (i.e. the sum of significance coefficients of the actually compromised sensors), ts = total sum (i.e. the sum of significance coefficients of all sensors), t = tolerance, rcn = reward for compromising the network, s = number of sensors, cps = cost per sensor, a = attacks, cpa = cost per attack, tc = tolerance cost [3].

Once the matrix is populated with the attacker’s payoff values, each node is faced with a challenging task of choosing the right action out of a set of predefined actions. Choosing the most beneficial action, which maximizes its utility regardless of the attacker’s next move, will be the Nash Equilibrium of this game. Nash Equilibrium is a set that consists of the strategies of all players, called optimal strategies, and that leads to a payoff for each player such that none of them can unilaterally change their strategy and gain higher reward than before [4].

We reapplied the same model, this time to opportunistic radio switching, whereby nodes opportunistically select their preferred transmission power and radio parameters with respect to the application requirements. In this scenario radio-mode-switching plays the role of a defender and Tx-switching is the attacker, which is synonymous with the region highlighted as the first important factor for defender’s strategy. In this game, one process (attacker) in LoRaBox attempts to increase the transmission power by increasing the Tx (table 2), as a result of analyzing the received packet’s properties through Wi-Fi (RSSI, SNR, Airtime). Another process attempts to avoid increasing Tx by forcing the higher LoRa modes (table 1). LoRaBox runs both processes concurrently.

IPM as shown in the schematic description in Figure 6 shows has two major differences from IDM:

1. the defender knows which sensors are attacked
2. the game is repeated for many rounds in a static from.

The goal of both attacker and defender is to compromise and defend the network with the least possible cost respectively. Based on this figure, the distribution of the means of the attacks defines the attacker’s strategies according to the number of recoveries and the number of sensors in each round. Therefore, the number of attacks and recoveries by both players are limited to the number of uncompromised and compromised nodes respectively.

\[
AP = ta \times (rcs - ac) + tr \times (rcps - rcs) + s \times ac + \left( \frac{cse}{Fs} \geq t \right) \times rcn
\]

(2)

Function 2 shows the attacker’s payoff where, ta = total attacks, rcs = reward for compromising a sensor, ac = attack cost, tr = total recoveries, rcp = recovery cost per sensor, rcs = reward for compromised sensor, sc = sensor cost, cse = compromised sensor at the end and tns = total number of sensors [3].

LoRaBox applies IPM when choosing LoRa mode based on their operational application requirement at a fixed Tx. In this game LoRaBox attempts to minimize the cost imposed on itself by estimating the total costs according to the previous packets and the application requirements. LoRaBox can ultimately choose the best set of modes for a given SNR, RSSI and Airtime, with the least overheads on its resources.

In this scenario, LoRaBox firstly conducts multiple rounds of auction by advertising the available subtasks (each sequence of the operation as described in figure 3). In each round, it estimates a bid using a different LoRa mode for the
advertised task by taking into account the estimated cost (energy expenditure) for executing that particular task. The estimated cost is based on:

1. RSSI for that given location (assuming that the node is static)
2. Airtime (direct impact on transceiver activity and its power consumption)
3. Number of transmission done via Wi-Fi for analysis (high energy consumption)
4. Packet loss

LoRaBox then applies IPM to the collected estimations and selects the best mode for executing the advertised task. What is important to note here is that unlike the IDM, in this scenario only maximizing each individual node’s utility is of importance and not the application requirements. Therefore, from the node’s point of view, the reward of this auction is spending the least processing time participating in analyzing the received packets via Wi-Fi.

![Figure 7: Auction for advertising a task to cluster-members](image)

Figure 7 shows an exemplar auction in which the LoRaBox estimates the utilities of LoRa modes V, X, Y, and Z using a fixed Tx, and analyzing four packets of A, B, C, and D. As this figure shows, packet D from LoRa mode Z responds four variable costs of 1-7-16-19 over 4 trials. Whereas packet D from LoRa mode V bids 1-5-20-9 for the same set of trials. Assuming that each LoRa mode in this figure represents a round of estimation in IPM, LoRaBox may decide which strategy results in higher utility (goal of IDM) and which Tx to allocate the task to where minimum energy expenditure is of concern.

### Table 3: Processing times

<table>
<thead>
<tr>
<th>Available Processing Time ($P_A$)</th>
<th>IDM</th>
<th>IPM</th>
</tr>
</thead>
<tbody>
<tr>
<td>10000</td>
<td>2000</td>
<td></td>
</tr>
<tr>
<td>Pre-deployed Task’s Required Processing Time ($P_R$)</td>
<td>5000</td>
<td>1000</td>
</tr>
<tr>
<td>New Task’s Processing Time ($P_N$)</td>
<td>5000</td>
<td>1000</td>
</tr>
<tr>
<td>Query/Response Processing Time ($P_Q$)</td>
<td>1000</td>
<td>200</td>
</tr>
<tr>
<td>Total Energy ($E_{total}$)</td>
<td>100,000</td>
<td>200,000</td>
</tr>
</tbody>
</table>

Applying the auction-based pricing equations [5-6] to the above-mentioned application, results in the following figures for the IDM and the IPM models in table 3. These figures are based on the number of uplink communication tasks with LoRa gateway via LoRa, and the number of downlink communication tasks for analysing the received packets via Wi-Fi.

It is worth noting that the actual processing time in has been defined in millisecond unit. However, for simplicity the above figures are normalised by a factor of 3000000. That is based on the LoRaBox’s 4000 mAh battery.

Once the utilities of different models, LoRa modes and communication tasks’ prices have been calculated using the aforementioned auction-based techniques, the rewards gained by LoRaBox, based on the notations given in table 3, can be calculated using the following functions:

$$E_{saving} = P_a - \left( \frac{\sum_{k=0}^{n} a_k}{n} + \frac{\sum_{j=0}^{m} p_{j0}}{m} + \frac{\sum_{k=0}^{m} r_k}{m} \right)$$  \hspace{1cm} (1)

This function simply returns how much energy can be saved by taking into account the number of pre-deployed tasks (n), new tasks (m) and the total number of query/responses ($k$).

### III. CASE STUDIES

We conducted a number of experiments in order to verify the effectiveness of the aforementioned game theoretic and auction-based approach in using the LoRaBox in the QEOG.

In our first experiment, we validate how effectively this approach contributes towards minimum airtime and packet loss with respect to application’s lifetime expectancy. We selected 30 points in the QEOG from which the LoRaBox transmits 20 packets in 4 different rounds:

1. Packets transmitted with no opportunistic algorithms using mode 10 (SF: 7, CR: 4/5, B/W: 500 KHz) and Tx: 7 dbm for maximum range, higher bandwidth and shorter airtime.
2. Round 1 with opportunistic algorithms
3. Packets transmitted with no opportunistic algorithms using mode 1 (SF: 12, CR: 4/5, B/W: 125 KHz) and Tx: 14 dbm for maximum range, low bandwidth and longer airtime.
4. Round 3 with with opportunistic algorithms

The first 2 rounds the application requirement does not allow the LoRaBox to exceed 7dbm for the transmission power as the lifetime of the system is set to 86,400,000 ms, whereas in round 3 and 4, there is no limit for both Tx and LoRa modes, as the system lifetime is not vital.

![Figure 8: Packets transmitted on mode 10](image)
locations didn’t reach the gateway (indicated by NS). The RSSI of the received packets were also quite low.

In the second round we used IPM, IDM and auctions to investigate whether they make any improvement to the RSSI by adjusting the SF and fixed power.

In round 2, As table 4 shows, 7 more locations (4,12,16,22-25) which had no signal in round 1, are now reporting good signals.

LoRaBox automatically increased the Tx from 0 (in round 1) to 7 (in round 2). However, this increase of Tx was not only meant for the locations, which had no reception in the first round. In this round LoRaBox polled all LoRa modes on blind spots sequentially in order to improve the signal.

| Table 4: Packets transmitted on mode 10 with opportunistic algorithms |
|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| RSSI | SNR | RSSI | SNR | RSSI | SNR | RSSI | SNR |
| 1 | -21 | -29.0 | 6 | -22.1 | 9 | -22.1 | 9 |
| 2 | -18 | -29.1 | 8 | -22.1 | 9 | -22.1 | 9 |
| 3 | -19 | -29.1 | 8 | -22.1 | 9 | -22.1 | 9 |
| 4 | -19 | -29.1 | 8 | -22.1 | 9 | -22.1 | 9 |
| 5 | -19 | -29.1 | 8 | -22.1 | 9 | -22.1 | 9 |

In round 3, data were transmitted at highest SF and Tx, using LoRa mode 1 and transmission power 14dBm.

| Table 5: Packets transmitted on mode 1 without opportunistic algorithms |
|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| RSSI | SNR | RSSI | SNR | RSSI | SNR | RSSI | SNR |
| 1 | -20 | -11.0 | 6 | -17.8 | 9 | -17.8 | 9 |
| 2 | -20 | -11.0 | 6 | -17.8 | 9 | -17.8 | 9 |
| 3 | -20 | -11.0 | 6 | -17.8 | 9 | -17.8 | 9 |
| 4 | -20 | -11.0 | 6 | -17.8 | 9 | -17.8 | 9 |
| 5 | -20 | -11.0 | 6 | -17.8 | 9 | -17.8 | 9 |

In the context of game theory, LoRaBox needs to weight its strategies and choose the best option, which maximises or at the least maintains its profit (utility) whilst meeting old and new applications’ requirements. As Table 6 shows, there is a significant drop in RSSI level after applying the algorithms. In this stage LoRaBox selects LoRa mode 4 and Tx level 10. In this scenario the system settles for this in order to minimise the packet loss as instructed by the application requirements.

Using both auction and game-theory, node’s utility however is defined by the amount of processing time spent on the given tasks, where maximising utility means spending less processing, thus saving more energy. Using opportunistic algorithms in round 4 resulted in nodes cut back on Tx by testing out different modes and Tx levels, in order to find an optimal configuration.

Therefore, as Figure 9 shows, IDM and IPM combined with auction, contributed to an average of between 2-9 dBm improvement in receiver’s sensitivity and packet RSSI based on each LoRa mode listed in table 1. As this figure shows, both approaches converge on LoRa modes 3,4 and 5 for maximizing node’s lifetime and meeting application requirements.

In this paper we have shown how utilizing game theory and auction-based algorithms can improve opportunistic radio communication of LPWA LoRa for various edge-processing applications, in which time-on-air, receiver sensitivity, packet loss and energy consumption can be optimised with respect to the application requirements.

ACKNOWLEDGEMENT

Special thanks to K Maraslis, T. Tryfonas and G. Oikonomou towards their collaborative works, with regards to the proposed IDM and IPM models in the previous publications.

REFERENCES


