Chapter 3

Centralized and distributed approaches to estimate the time duration for effective use of the Spreading factors

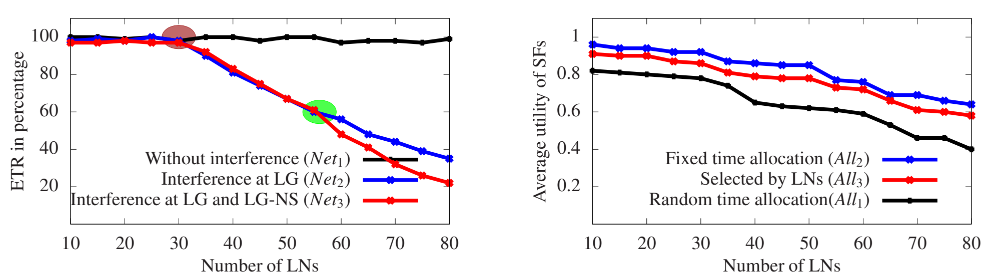
3.1 Introduction

The LoRa is one of the most promising wireless communication technology because it allows for long-range communication with low power consumption. The interference problem in LoRa occurs when LNs are attached with a LG using the same spreading factor. To overcome the problem, the literature assumes that the SF is allocated to each node for a fixed time duration. Since all LNs may not have equal data to transmit to the LG, the network revenue estimated by using fixed time duration is lower than the allocation based on the size of the data of the LNs. In this chapter, we assume that each LN in the network takes multiple services from the same or different LGs. We address the problem: *How long a LN uses the allocated SF from a given LG such that the LN satisfies its service requirement and the network maximizes its revenue?* We refer to this problem as (n,m,s,c)-Time duration Allocation problem or simply (n,m,s,c)-TA problem, where n, m, s, and c denote the LN, LG, service, and CR, respectively. Presence of multiple LNs in the network with selfish behaviour and limited resources for the data transmission leads to the competition among LNs. Therefore, we use the game-theoretic framework for solving the above problem.

3.1.1 Motivation of this work

The work in this chapter is motivated by the following observations from the literature.

• *Effective transmission rate*: The literature in LoRa network did not consider the effects of the interference problem from EUs to the NS while analyzing the performance of the network. Part (a) of Figure 3.1 illustrates the Effective Transmission Rate (ETR) in percentage which is the ratio of the achieved Transmission Rate (TR) with interference to the TR without any interference problem. The authors in [16, 41] assumed that the ETR equals to the TR without any interference problem as shown Net_1. Authors in [14, 18] analyzed the SFs interference problem by considering the TR only between LN and LG. Therefore, the ETR in [18] goes down when large number of LNs are connected as shown Net_2. The more realistic ETR is shown as Net_3 in part (a) of Figure 3.1, where TR goes down due to the interference at LGs and insufficient bandwidth between LG and NS.



(a) Impact of LNs on ETR. (b) Impact of LNs on utility of SFs.

Figure 3.1: Motivations of the proposed work.

• *Allocation of time duration for using the SFs*: The authors in [15, 41, 42] assumed that each LN directly interacts with the LG to setup the network. Part (b) of Figure 3.1 illustrates the average utility of the allocated SFs which is calculated as $1 - \left(1/N \sum_{i=1}^{N} \frac{(t_a^i - t_u^i)}{t_a^i}\right)$, where t_a^i and t_u^i are the allocated and used time duration of SFs by a LN i, respectively, and $1 \leq i \leq N$. It shows that if the LG randomly allocates time duration to the LNs for using the SFs then some LNs do not access or over access the network and therefore average utility goes down (All_1 in figure) [42]. Figure also illustrates that the fixed time duration allocation of SFs to each LN (All_2 in figure) reduces the average utility [15,41]. Also due to imbalance of load on SFs, average utility goes down (All_3 in figure).

• *Connectivity of LNs to the LGs*: Authors in [39, 43, 44] demonstrated that network performance drastically reduces in a densely deployed LNs network due to the imbalance of load on the LGs.

3.1.2 Major contributions

This work addresses (n,m,s,c)-TA problem for LoRa network. Apart from this, the major contributions are as follows:

• *Effective transmission rate*: This work considers a network with multiple LGs, where each LN takes multiple services from the same or different LGs. In such scenario, transmission rate between LGs and NS is also an important aspect to be considered while selection of LG. We estimate the *effective transmission rate* between a LN and the NS for data transmission.

• *Nash Equilibrium among LNs*: We formulate the interaction among the LNs as a *Nash equilibrium game* to allocate the time duration of using the SFs [28]. We propose a NE algorithm for finding the optimal time duration for all LNs to use the given SF.

• *Stackelberg Game between LGs and LNs*: Imbalance of the load on LGs leads to the use of same SF by more than one LN which causes the interference problem. After finding NE among LNs, we formulate the interaction between LGs and LNs as a Stackelberg game to balance the load on the LGs and reduce the effect of the interference problem.

We also provide the proof of the existence and uniqueness of the SE for the game.

• *Implementation of the solution*: We propose distributed and centralized algorithms based on the participation of LNs and the selection of processing node, respectively.

• Finally, we validate the analysis and demonstrate the significance of the number of LNs, LGs, services, and transmission rate in the estimation of time duration for using the SFs.

The rest of the chapter is organized as follows: Next section describes the network model and problem statement. The game model and the analysis for solving the (n,m,s,c)-TA problem are presented in Sections 3.3 and 3.4, respectively. Next, Section 3.5 illustrates how to implement the proposed solution in LoRa network. Finally, Section 3.6 presents the results followed by the conclusions in Section 3.7.

3.2 Preliminaries and problem statement

This section describes the network model, definition of effective transmission rate, and the problem statement. Throughout this chapter, we use the bold notation to indicate a vector.

3.2.1 Network model

We consider a scenario where each LN in network transmits data to LG over a single wireless hop and LG transmits it to NS through a non-LoRa network as shown in Figure 3.2. BW in the network is the frequency range of the chirp signal. LoRa operates with SFs ranging from SF_7 to SF_{12}, where SF_7 and SF_{12} are the shortest and longest time on air, respectively. LoRa modulation adds forward error correction bits in every data transmission. This implementation is done by encoding 4-bit data with redundancies into l-bits, where $5 \leq l \leq 8$. The CR of network equals to $4/l$.

The LoRa network consists of set $\mathcal{N}$ of N LNs and set $\mathcal{M}$ of M LGs. Let n, m, s, and c denote the indexes of the LN, LG, service, and, CR, respectively. Each randomly

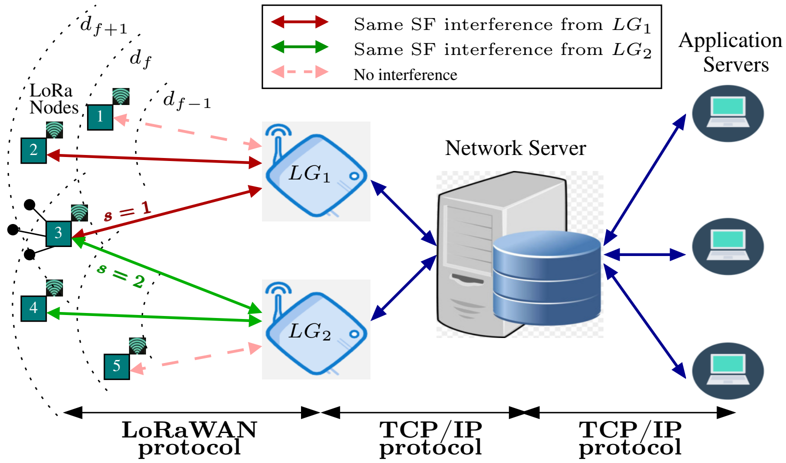


Figure 3.2: End-to-end architecture of a LoRa network.

deployed LN has sets $\mathcal{S} = \{1, \cdots, S\}$ and $\mathcal{C} = \{1, \cdots, C\}$, where S and C denote the number of services and CRs, respectively. Let $t_{n,m}^{s,c}$ is the time duration of a LN $n \in \mathcal{N}$ using SF from $m \in \mathcal{M}$ for $s \in \mathcal{S}$ on $c \in \mathcal{C}$. The service time vector of all LNs for given s, c, and m is therefore

$$\mathbf{t}_m^{s,c} = [t_{1,m}^{s,c}, \cdots, t_{n,m}^{s,c}, \cdots, t_{N,m}^{s,c}]^T. \tag{3.1}$$

Let $\rho_m^{s,c}$ is the price per unit time paid by a LN to m for s on c. The price vector of all LGs for a service s on c is given by

$$\boldsymbol{\rho}^{s,c} = [\rho_1^{s,c}, \cdots, \rho_m^{s,c}, \cdots, \rho_M^{s,c}]^T. \tag{3.2}$$

3.2.2 Effective transmission rate

The Transmission Rate (TR) in LoRa is the number of bits successfully transmitted per unit time from a LN to the NS.

Definition 3.1 *Let a LN $n \in \mathcal{N}$ for $s \in \mathcal{S}$ and $c \in \mathcal{C}$ uses SF f and BW W for transmitting the data to LG $m \in \mathcal{M}$. The transmission rate from n to m is given* *by [45]*

$$r_{n,m}^{s,c} = W \times \frac{f}{2^f} \times \frac{4}{4+c}. \tag{3.3}$$

Let N_m LNs lie in the range of $m \in \mathcal{M}$. Figure 3.2 illustrates that a LN lies in the range of two LGs. In such scenario, multiple LNs can simultaneously transfer data to the same LG with same SF and create interference problem in the network. However, LoRa network does not have interference problem when LNs are connected with different LGs even using the same SF. Equation 3.4 describes the probability of successful transmission of a bit from LN n to the LG m [18],

$$P_{succ}(r_n^m) = \sum_{j=1}^{N_m} \binom{N_m}{j} (1-p_f)^{N_m - j} \times P^n_{cap_{coSF}}(j), \tag{3.4}$$

where j denotes the total number of LNs using SF f among N_m, p_f is the probability of getting SF f, and $P^n_{cap_{coSF}}(j)$ is the probability that a received signal from n has a SNR above the threshold. The effective TR from n to m, denoted by $\bar{r}_{n,m}^{s,c}$, is the successful transmission of the data per unit time from LN n to the LG m, *i.e.*,

$$\bar{r}_{n,m}^{s,c} = r_{n,m}^{s,c} \times P_{succ}(r_n^m). \tag{3.5}$$

Next, the network transfers the data from LG to NS using non-LoRa network. Let $\bar{\mathfrak{r}}^m$ denotes the TR from m to NS. The effective TR from m to NS by using Shannon-Hartley equation [46] is given as

$$\bar{\mathfrak{r}}^m = b^m \log_2 \left(1 + \frac{g^m \mathfrak{p}^m}{\sigma^2}\right), \tag{3.6}$$

where b^m, g^m, $\mathfrak{p}^m$, and σ^2 are the bandwidth, channel gain, power used, and white gaussian noise in the network from LG m to NS, respectively.

Definition 3.2 *The effective transmission rate of the end-to-end network (i.e., n to m*

using LoRa network and m to NS using other network) is given as

$$R_{n,m}^{s,c} = \min\left(\bar{r}_{n,m}^{s,c}, \bar{\mathfrak{r}}^{m}\right), \tag{3.7}$$

where $\bar{r}_{n,m}^{s,c}$ and $\bar{\mathfrak{r}}^{m}$ are given in Equations 3.5 and 3.6, respectively.

3.2.3 Overview of (n,m,s,c)-TA problem

In this chapter, we deal with LoRa network where a LN n can learn the price of LG m for a given s and c, and adjust its service time duration for using the allocated SF to maximize its utility. We are interested in determining the service price of the LG for allocating the SF to the connected LNs such that its utility is maximized. We consider SG for solving the problem where LGs and LNs work as leaders and followers, respectively. The LGs start the game and choose the price of the received data from the LNs. Based on the LG strategy, each LN selects the time duration for transferring the data to the LG using allocated SF which maximizes its utility.

3.3 Game model for (n,m,s,c)-TA problem

This section derives the expressions of the models and formulates the optimization problems of followers and leaders.

3.3.1 Follower (LoRa node) model

Each LG $m \in \mathcal{M}$ at the beginning of the game announces the price $\rho_{m}^{s,c}$ as given in Equation 3.2. Based on $\rho_{m}^{s,c}$, each LN reacts by selecting an optimal strategy which maximizes its utility.

• **Price gain from end users:** The LNs gain price from the EUs for transmitting the data to the NS in a given time duration. Due to the interference problem, the gain of a LN also depends on the strategies of other LNs. To enhance the utility, the LNs try to transfer more data and deviate from the stable state. Hence to prevent LNs from deviation, price gain function reduces price when any LN transmits data for more time duration than the time duration estimated by the SE. Therefore, the price function for the given values of $m \in \mathcal{M}$, $s \in \mathcal{S}$, and $c \in \mathcal{C}$, is modeled as oligopoly market [28] and defined as,

$$\mathcal{P}(\mathbf{t}_m^{s,c}) = a - \sum_{i=1}^{N} R_{i,m}^{s,c} t_{i,m}^{s,c}, \tag{3.8}$$

where $\mathbf{t}_m^{s,c} = [t_{1,m}^{s,c}, \cdots, t_{n,m}^{s,c} \cdots, t_{N,m}^{s,c}]^T$ and a is the forecasted demand of total data. Such forecasted demand is calculated based on the past demand of the transmitted data from all the LNs to the NS via LG and the available resources of the NS in the network. Since the magnitude of the price function can not be negative, we model this function as ReLu function given as:

$$\mathcal{P}'(\mathbf{t}_m^{s,c}) = \max\{0, \mathcal{P}(\mathbf{t}_m^{s,c})\}. \tag{3.9}$$

The price gain of a LN $n \in \mathcal{N}$ is the product of price function, effective transmission rate, and data transmission time, *i.e.*,

$$L_g(\mathbf{t}_n) = \sum_{s=1}^{S} \sum_{c=1}^{C} \sum_{m=1}^{M} \mathcal{P}'(\mathbf{t}_m^{s,c}) R_{n,m}^{s,c} t_{n,m}^{s,c}, \tag{3.10}$$

where, $\mathbf{t}_n = [\mathbf{t}_n^1, \cdots, \mathbf{t}_n^s, \cdots, \mathbf{t}_n^S]$, $\mathbf{t}_n^s = [\mathbf{t}_n^{s,1}, \cdots, \mathbf{t}_n^{s,c}, \cdots, \mathbf{t}_n^{s,C}]$, and $\mathbf{t}_n^{s,c} = [t_{n,1}^{s,c}, \cdots, t_{n,m}^{s,c}, \cdots, t_{n,M}^{s,c}]$.

• **Price paid to the LGs:** The LNs need to pay the price to LGs for relaying the data to the NS. The price paid by $n \in \mathcal{N}$ is the product of the time it uses the LGs for relaying the data and price per unit time charge by LGs, *i.e.*,

$$L_c(\mathbf{t}_n) = \delta_n \sum_{s=1}^{S} \sum_{c=1}^{C} \sum_{m=1}^{M} t_{n,m}^{s,c} \rho_m^{s,c}, \tag{3.11}$$

where δ_n is the constant which makes the cost to the same order of magnitude as the utility.

3.3.1.1 Utility function of follower

The utility of a follower $n \in \mathcal{N}$ during $\mathbf{t}_n$ is the difference of the price gained from the EUs and the cost paid to the LGs, *i.e.*,

$$\begin{aligned} U_n(\mathbf{t}_n, \mathbf{t}_{-n}) &= L_g(\mathbf{t}_n) - L_c(\mathbf{t}_n), \\ &= \sum_{s=1}^{S}\sum_{c=1}^{C}\sum_{m=1}^{M}\left(\left(a - \sum_{i=1}^{N} R_{i,m}^{s,c} t_{i,m}^{s,c}\right) R_{n,m}^{s,c} t_{n,m}^{s,c} - \delta_n t_{n,m}^{s,c} \rho_m^{s,c}\right). \end{aligned} \tag{3.12}$$

3.3.1.2 Follower game

Based on the price announced by the leaders, the followers compete with each other to maximize their utilities. The game uses the following information:

- **Players:** Each LN is one player and there are N players.
- **Strategies:** The strategy of a follower n is the time duration vector $\mathbf{t}_n$ and total service time of n for s does not exceed its remaining duty cycle $T_{n,s}^{max}$.
- **Utilities:** Let $\mathbf{t}_{-n}$ is the strategies vector for all the players except of player n. The utility of n is given in Equation 3.12.

The follower level game is expressed as follows:

$$\begin{aligned} \textbf{Problem 1} \qquad \max_{\mathbf{t}_n} \quad & U_n(\mathbf{t}_n, \mathbf{t}_{-n}), \\ \text{s.t.} \quad & \sum_{c=1}^{C}\sum_{m=1}^{M} t_{n,m}^{s,c} \le T_{n,s}^{max}, \\ & t_{n,m}^{s,c} \ge 0, \\ \text{where } & \forall s \in \mathcal{S}, \forall c \in \mathcal{C}, \forall m \in \mathcal{M}. \end{aligned} \tag{3.13}$$

3.3.2 Leader (LoRa gateway) model

Due to imbalance of load on SFs, interference problem occurs in the network. The leaders set the pricing strategies in such a way that the SFs can be allocated to the LNs in a balanced way. The utility function of the leader consists the following terms:

• **Price gain from the LNs:** The LGs gain the prices from the LNs for providing services as given in Equation 3.11. The price gain from LNs during $\mathbf{t}_m$ time duration is given as

$$G_g(\mathbf{t}_m) = \sum_{n=1}^{N}\sum_{s=1}^{S}\sum_{c=1}^{C} \rho_m^{s,c} t_{n,m}^{s,c}, \tag{3.14}$$

where $\mathbf{t}_m = [\mathbf{t}_{1,m}, \cdots, \mathbf{t}_{n,m}, \cdots, \mathbf{t}_{N,m}]$, $\mathbf{t}_{n,m} = [\mathbf{t}_{n,m}^1, \cdots, \mathbf{t}_{n,m}^s, \cdots, \mathbf{t}_{n,m}^S]$, and $\mathbf{t}_{n,m}^s = [t_{n,m}^{s,1}, \cdots, t_{n,m}^{s,c}, \cdots, t_{n,m}^{s,C}]$.

• **Maintenance cost of the LG:** Let x_m be the cost for providing services to the LNs for $m \in \mathcal{M}$. Therefore the maintenance cost of m during $\mathbf{t}_m$ is given as

$$G_c(\mathbf{t}_m) = x_m \sum_{n=1}^{N}\sum_{s=1}^{S}\sum_{c=1}^{C} t_{n,m}^{s,c}. \tag{3.15}$$

3.3.2.1 Utility function of leader

The net utility of a leader $m \in \mathcal{M}$ when it charges $\boldsymbol{\rho}_m$ price is the difference of the price gained from the LNs and the maintenance cost, *i.e.,*

$$\begin{aligned} U_m(\boldsymbol{\rho}_m, \boldsymbol{\rho}_{-m}) =& G_g(\mathbf{t}_m) - G_c(\mathbf{t}_m), \\ =& \sum_{n=1}^{N}\sum_{s=1}^{S}\sum_{c=1}^{C} \rho_m^{s,c} t_{n,m}^{s,c} - x_m \sum_{n=1}^{N}\sum_{s=1}^{S}\sum_{c=1}^{C} t_{n,m}^{s,c}. \end{aligned} \tag{3.16}$$

3.3.2.2 Leader game

Based on the strategies selected by followers, the leaders optimize their strategies to maximize their utilities. The game uses the following information:

- **Players:** Each LG is one player and there are M players.
- **Strategies:** The price $\rho_m^{s,c}$ paid by the LNs for the data and time duration taken by n for all services on m must not exceed its allocated time $T_{n,m}^{max}$.
- **Utilities:** The utility for a leader is given in Equation 3.16.

The leader level game is expressed as follows:

$$\textbf{Problem 2} \quad \max_{\boldsymbol{\rho}_m} \quad U_m(\boldsymbol{\rho}_m, \boldsymbol{\rho}_{-m}),$$
$$s.t. \quad \sum_{s=1}^{S} \sum_{c=1}^{C} t_{n,m}^{s,c} \leq T_{n,m}^{max},$$
$$\rho_m^{s,c} \geq 0,$$
$$\text{where } \forall s \in \mathcal{S}, \forall c \in \mathcal{C}, \forall n \in \mathcal{N}. \tag{3.17}$$

3.4 Game analysis for solving (n,m,s,c)-TA

This section solves (n,m,s,c)-TA problem by using the Stackelberg game. Figure 3.3 illustrates the block diagram of Stackelberg Game.

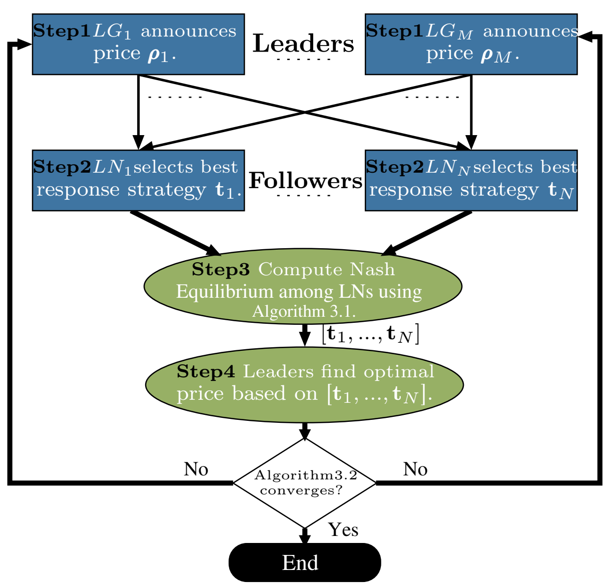


Figure 3.3: Illustration of the block diagram of Stackelberg game.

3.4.1 Best response strategy of LNs

To define the best response strategies of LNs, we solve **Problem 1** which maximizes the utilities of LNs with the constraints. For simplicity, let $t^{s,c}_{n,m}$ is denoted as $\hat{t}_n$, $R^{s,c}_{n,m}$ as $\hat{R}_n$, and $\rho^{s,c}_m$ as $\hat{\rho}_m$. Theorem 3.1 proves the existence of the best response strategy of each LN.

Theorem 3.1 *Let $\hat{t}_n$ be the strategy of a LN $n \in \mathcal{N}$ for accessing the service $s \in \mathcal{S}$ from LG $m \in \mathcal{M}$ on CR $c \in \mathcal{C}$. The best response $\hat{t}^\star_n$ of the LN is given as*

$$\hat{t}^\star_n = \frac{1}{2\hat{R}^2_n}\left(\hat{Q}_n - \sum_{c=1}^{C}\sum_{m=1}^{M} 2\hat{R}^2_n\left(\sum_{c=1}^{C}\sum_{m=1}^{M}\frac{\hat{Q}_n}{2\hat{R}^2_n} - T^{max}_{n,s}\right)\right),$$

where

$$\hat{Q}_n = a\hat{R}_n - \hat{R}_n\sum_{j=1, j\neq n}^{N}\hat{R}_j\hat{t}_j - \delta_n\hat{\rho}_m. \tag{3.18}$$

Proof: We start with a simple scenario where $M = 2$, $S = 2$, and $C = 1$ and will generalize the scenario later in this section. From Equation 3.13, the utility of a LN $n \in \mathcal{N}$ is given as

$$\begin{aligned} \max_{\mathbf{t}_n} \quad & U_n(\mathbf{t}_n, \mathbf{t}_{-n}) = \sum_{s=1}^{2}\sum_{m=1}^{2}\left(a - \sum_{i=1}^{N}\hat{R}_i\hat{t}_i\right)\hat{R}_n\hat{t}_n - \delta_n\sum_{s=1}^{2}\sum_{m=1}^{2}\hat{t}_n\hat{\rho}_m, \\ \text{s.t.} \quad & t^{1,1}_{n,1}, t^{2,1}_{n,1}, t^{1,1}_{n,2}, t^{2,1}_{n,2} \geq 0, \\ & \sum_{m=1}^{2}\hat{t}_n \leq T^{max}_{n,s}. \end{aligned} \tag{3.19}$$

Using Lagrangian multipliers $\lambda_{n,1}, \lambda_{n,2}, \lambda_{n,3}, \lambda_{n,4}$ and $\lambda_{n,5}$ for constraints defined in

Equation 3.19

$$\mathcal{L}_n(\mathbf{t}_n, \mathbf{t}_{-n}) = \sum_{s=1}^{2}\sum_{m=1}^{2}\left(a - \sum_{i=1}^{N}\hat{R}_i\hat{t}_i\right)\hat{R}_n\hat{t}_n - \delta_n\sum_{s=1}^{2}\sum_{m=1}^{2}\hat{t}_n\hat{\rho}_m - \lambda_{n,1}\left(\sum_{m=1}^{2}\hat{t}_n - T_{n,s}^{max}\right)$$
$$+\lambda_{n,2}t_{n,1}^{1,1} + \lambda_{n,3}t_{n,1}^{2,1} + \lambda_{n,4}t_{n,2}^{1,1} + \lambda_{n,5}t_{n,2}^{2,1},$$
$$\text{s.t.} \quad \lambda_{n,1}(\sum_{m=1}^{2}\hat{t}_n - T_{n,s}^{max}) = 0, \lambda_{n,1} > 0, \lambda_{n,2}t_{n,1}^{1,1}, \lambda_{n,3}t_{n,1}^{2,1}, \lambda_{n,4}t_{n,2}^{1,1}, \lambda_{n,5}t_{n,2}^{2,1} = 0,$$
$$\forall l \in \{2,3,4,5\}, \lambda_{n,l}, t_{n,1}^{1,1}, t_{n,1}^{2,1}, t_{n,2}^{1,1}, t_{n,2}^{2,1} \geq 0. \tag{3.20}$$

The first order derivative of $\mathcal{L}_n(\mathbf{t}_n, \mathbf{t}_{-n})$ *w.r.t.* $t_{n,1}^{1,1}$ is $\frac{d\mathcal{L}_n(\mathbf{t}_n,\mathbf{t}_{-n})}{dt_{n,1}^{1,1}} = aR_{n,1}^{1,1} - 2(R_{n,1}^{1,1})^2t_{n,1}^{1,1} - R_{n,1}^{1,1}\sum_{j=1,j\neq n}^{N}R_{j,1}^{1,1}t_{j,1}^{1,1} - \delta_n\rho_1^{1,1} - \lambda_{n,1} + \lambda_{n,2}$. The second order derivative of $\mathcal{L}_n(\mathbf{t}_n, \mathbf{t}_{-n})$ *w.r.t.* $t_{n,1}^{1,1}$ is $-2(R_{n,1}^{1,1})^2$. Similar, we can derived the derivatives of $\mathcal{L}_n(\mathbf{t}_n, \mathbf{t}_{-n})$ *w.r.t.* $t_{n,1}^{2,1}, t_{n,2}^{1,1}$ and $t_{n,2}^{2,1}$ and conclude that $\frac{d^2\mathcal{L}_n(\mathbf{t}_n,\mathbf{t}_{-n})}{d(\hat{t}_n)^2} < 0$. It shows that the second order derivative of $\mathcal{L}_n(\mathbf{t}_n, \mathbf{t}_{-n})$ is negative, the utility function in Equation 3.13 is concave and continuous, therefore, the follower level game has at least one SE. Let A and B are the coefficient matrix. $\frac{d\mathcal{L}_n(\mathbf{t}_n,\mathbf{t}_{-n})}{dt_{n,1}^{1,1}}$ can be rewritten as

$$\overbrace{\begin{bmatrix} A_{1,1} & A_{1,2} & \dots & A_{1,N} \\ A_{2,1} & A_{2,2} & \dots & A_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ A_{N,1} & A_{N,2} & \dots & A_{N,N} \end{bmatrix}}^{A}\overbrace{\begin{bmatrix} t_{1,1}^{1,1} \\ t_{2,1}^{1,1} \\ \vdots \\ t_{N,1}^{1,1} \end{bmatrix}}^{\mathbf{t}} = \overbrace{\begin{bmatrix} b'_1 - \delta_1\rho_1^{1,1} \\ b'_2 - \delta_2\rho_1^{1,1} \\ \vdots \\ b'_N - \delta_N\rho_1^{1,1} \end{bmatrix}}^{B}, \tag{3.21}$$

where $A_{n,n} = 2(R_{n,1}^{1,1})$, $A_{n,l} = R_{n,1}^{1,1}R_{l,1}^{1,1}$ for $l \in \mathcal{N}$ and $l \neq n$, and $b'_n = aR_{n,1}^{1,1} - \lambda_{n,1} + \lambda_{n,2}$. From the strictly diagonal dominant theorem [28], the matrix A is nonsingular if $(R_{n,1}^{1,1})^2 \geq \frac{R_{n,1}^{1,1}\sum_{j=1,j\neq n}^{N}R_{j,1}^{1,1}}{2}, \forall n \in \mathcal{N}$. Since matrix A is nonsingular, inverse of the matrix A is possible. Best response strategies of the followers as the time duration $\hat{\mathbf{t}} = [\hat{t}_1, \cdots, \hat{t}_n, \cdots, \hat{t}_N]$ to transmit data to the LG can be calculated as $A^{-1}B$, which is defined as $t^{w,1\star}_{n,v} = \frac{aR^{w,1}_{n,v} - R^{w,1}_{n,v}\sum_{j\neq n} R^{w,1}_{j,v} t^{w,1}_{j,v} - \delta_n \rho^{w,1}_v - \lambda_{n,1} + \lambda_{n,k}}{2(R^{w,1}_{n,v})^2}$, where $v, w \in \{1, 2\}$ and $k = \{2, 3, 4, 5\}$ for $(v, w) = \{(1, 1), (1, 2), (2, 1), (2, 2)\}$. The optimal time duration of n from m for s and c, for generating the network scenario where N LNs, M LGs, S services, and C CR exist, can be obtained from the constraint of Equation 3.20 as

$$\hat{t}^\star_n = \frac{a\hat{R}_n - \hat{R}_n \sum_{j=1,j\neq n}^{N} \hat{R}_j \hat{t}_j - \delta_n \hat{\rho}_m - \lambda_{n,1}}{2(\hat{R}_n)^2}. \tag{3.22}$$

By putting value of $\hat{t}_n$ into the constraint of Equation 3.20, we get

$$\lambda_{n,1} = \sum_{c=1}^{C}\sum_{m=1}^{M} 2\hat{R}^2_n \left(\sum_{c=1}^{C}\sum_{m=1}^{M} \frac{a\hat{R}_n - \hat{R}_n \sum_{j=1,j\neq n}^{N} \hat{R}_j \hat{t}_j - \delta_n \hat{\rho}_m}{2\hat{R}^2_n} - T^{max}_{n,s} \right). \tag{3.23}$$

Substituting $\lambda_{n,1}$ from Equation 3.23 into Equation 3.22 and hence proved. □

Proposition 1 *The best response strategy of a LN $n \in \mathcal{N}$ is unique and optimal solution.*

Proof: (**Proof of uniqueness**) The best response strategy of a follower, defined in Equation 3.22, is unique if it consists *positivity*, *monotonicity*, and *scalability*. From the constraint of Equation 3.13, we confirm that $\hat{t}_n > 0$ because $\sum_{c=1}^{C}\sum_{m=1}^{M} \frac{\hat{Q}_n}{2\hat{R}^2_n} - T^{max}_{n,s}$ will always return negative value which yields positivity of the $\hat{t}_n$. Let $\hat{t}_{-n}$ and $\hat{\hat{t}}_{-n}$ are the data transmission time duration of LNs except n. Equation 3.18 shows that if $\hat{t}_{-n} \geq \hat{\hat{t}}_{-n}$ then $\mathscr{F}(\hat{t}_{-n}) \leq \mathscr{F}(\hat{\hat{t}}_{-n})$ and therefore $\hat{t}_n$ consists monotonicity. The best response strategy $\hat{t}_n$ consists scalability if $\mu\mathscr{F}(\hat{t}_{-n})$ is greater than $\mathscr{F}(\mu\hat{t}_{-n})$ and $\mu \geq 0$. Using Equation 3.18, we get $\mu\mathscr{F}(\hat{t}_{-n}) - \mathscr{F}(\mu\hat{t}_{-n}) \geq 0$ and therefore scalability is proved. (**Proof of optimality**) Equation 3.13 is convex optimization problem because objective function $U_n(\mathbf{t}_n, \mathbf{t}_{-n})$, defined in Equation 3.22, is the quadratic function with respect to $\hat{t}_n$ and constraint is affine. From Equation 3.22, $\mathbf{t}^*_n = \min\{\mathbf{t}_n, T^{max}_{n,s}\}$. Let assume $\mathbf{t}_n \geq T^{max}_{n,s}$ and hence $\mathbf{t}^*_n = T^{max}_{n,s}$. The objective function $U_m(\mathbf{t}_n, \mathbf{t}_{-n})$, defined in Equation 3.16, after substituting $\mathbf{t}^*_n$ is to maximize **Problem 2**. We can observe that the objective function is an increasing function with respect to $\hat{\rho}_m$. For maximizing this function, we must have $\hat{\rho}_m \geq \frac{a\hat{R}_n - \hat{R}_n\mu\sum_{j\neq n}\hat{R}_j\hat{t}_j - \lambda_{n,1} - 2\hat{R}_n^2 T_{n,s}^{max}}{\delta_n}$. But from the assumption *i.e.,* $\mathbf{t}_n \geq T_{n,s}^{max}$, we have $\hat{\rho}_m \leq \frac{a\hat{R}_n - \hat{R}_n\mu\sum_{j\neq n}\hat{R}_j\hat{t}_j - \lambda_{n,1} - 2\hat{R}_n^2 T_{n,s}^{max}}{\delta_n}$, which is the contradiction. Thus $\mathbf{t}_n^* = \mathbf{t}_n$, which completes the proof.

□

3.4.2 Near Nash Equilibrium (NE) among the LNs

Consider a game of N players. A set of strategies $(\hat{t}_1, \ldots, \hat{t}_n, \ldots, \hat{t}_N)$ constitute NE if for $n \in \mathcal{N}$

$$U_n(\hat{t}_n^\star, \hat{t}_{-n}^\star) \geq U_n(\hat{t}_n, \hat{t}_{-n}^\star). \tag{3.24}$$

We propose a near optimal time duration allocation algorithm (Algorithm 3.1) that terminates in polynomial time. Algorithm 3.1 finds near NE by using the estimated best response strategies of LNs. We first define near NE (*i.e., η-NE*) as

Definition 3.3 *(η-NE): A set of strategies $(\hat{t}_1, \ldots, \hat{t}_n, \ldots, \hat{t}_N)$ constitute an η-NE of game if for $n \in \mathcal{N}$*

$$U_n(\hat{t}_n^\star, \hat{t}_{-n}^\star) \geq U_n(\hat{t}_n, \hat{t}_{-n}^\star) - \eta. \tag{3.25}$$

In η-NE, players have small incentive to deviate from NE but they can not increase their utilities by more than $\eta > 0$. Algorithm 3.1 will eventually converge to a fixed point, where strategies of LNs can not increase the utility by more than η. Algorithm 3.1 does not consider the communication overhead of announcing the prices because it is same for all LNs and thus does not has an impact on the utility in the game.

Theorem 3.2 *Near NE algorithm among LNs illustrated in Algorithm 3.1 reaches the η-NE in $O(N/\eta)$ iterations for any given $\eta > 0$.*

Proof: According to Definition 3.3, LN n changes its strategy if utility increases by more than η in each iteration. Thus $U_n(\hat{t}_n^{(\tau)}, \hat{t}_{-n}) - U_n(\hat{t}_n^{(\tau-1)}, \hat{t}_{-n}) > \eta$. Therefore, each

Algorithm 3.1: Near Nash Equilibrium among LNs

Input: Precision threshold η, $\tau \leftarrow 0$, $\hat{t}_n[0] \leftarrow \eta$;
Output: Best response strategy $\hat{t}_n$ of n ;

do
  $U_{n\tau}(\hat{t}_n, \hat{t}^{\star}_{-n}) = U_n(\hat{t}_n, \hat{t}^{\star}_{-n})$ using $\hat{t}_n[\tau]$;
  $\tau \leftarrow \tau + 1$; ▷ Using Equation 3.18 for estimating $\hat{t}_n[\tau + 1]$.
  $\hat{t}_n[\tau + 1] = \frac{a\hat{R}_n - \hat{R}_n \sum_{j=1, j\neq n}^{N} \hat{R}_j \hat{t}_j[\tau] - \delta_n \hat{\rho}_m - \lambda_{n,1}}{2(\hat{R}_n)^2}$;
  $U_{n\tau+1}(\hat{t}_n, \hat{t}^{\star}_{-n}) = U_n(\hat{t}_n, \hat{t}^{\star}_{-n})$ using $\hat{t}_n[\tau + 1]$;
while $\left(\left\|U_{n\tau+1}(\hat{t}_n, \hat{t}^{\star}_{-n}) - U_{n\tau}(\hat{t}_n, \hat{t}^{\star}_{-n})\right\| > \eta\right)$;

LN increases its utility by atleast η in each iteration. We next show the upper bound of the utility of LN at NE. An upper bound of the utility of each LN can be obtained when there is no competition with other LNs. As a result, utility of LN at NE is always less than $U_n(\hat{t}^{\star}_n, 0)$. Therefore, the number of iterations of LN n is atmost $\frac{U_n(\hat{t}^{\star}_n, 0)}{\eta}$. In a worst case, the number of iterations for any LN is no more than $\frac{\max_{n\in\mathcal{N}} U_n(\hat{t}^{\star}_n, 0)}{\eta}$. We can conclude that the time complexity of Algorithm 3.1 to determine near NE for all LNs is $O(N/\eta)$, and hence proved. □

3.4.3 Optimal strategies of LGs

In this section, we prove that the unique and optimal best response strategies of LNs is admitted by each LG. Using backward induction method, **Problem 2** can be rewritten as

$$
\begin{aligned}
\textbf{Problem 3:} \quad & \max_{\boldsymbol{\rho}_m} \quad U_m(\boldsymbol{\rho}_m, \mathbf{t}_n) = \sum_{n=1}^{N}\sum_{s=1}^{S}\sum_{c=1}^{C} \left(\hat{\rho}_m \hat{t}^{\star}_n - x_m \hat{t}^{\star}_n\right), \\
& s.t. \quad \sum_{s=1}^{S}\sum_{c=1}^{C} \hat{t}^{\star}_n \leq T^{max}_{n,m} \\
& \qquad \hat{\rho}_m \geq 0, \forall n \in \mathcal{N}, \forall m \in \mathcal{M}. \qquad (3.26)
\end{aligned}
$$

The optimal solution of **Problem 3** can be obtain by Karush-Kuhn-Tucker conditions and the Lagrangian is given as

$$\mathcal{L}_m(\boldsymbol{\rho}_m, \mathbf{t}_n) = \sum_{n=1}^{N}\sum_{s=1}^{S}\sum_{c=1}^{C} \hat{\rho}_m \hat{t}_n^{\star} - x_m \sum_{n=1}^{N}\sum_{s=1}^{S}\sum_{c=1}^{C} \hat{t}_n^{\star} - \Lambda_1 \sum_{s=1}^{S}\sum_{c=1}^{C}(\hat{t}_n^{\star} - T_{n,m}^{max}) + \Lambda_2 \hat{\rho}_m,$$
$$\text{s.t.} \quad \Lambda_1 \sum_{s=1}^{S}\sum_{c=1}^{C}(\hat{t}_n^{\star} - T_{n,m}^{max}) = 0, \{\Lambda_1, \Lambda_2\} \geq 0, \text{ and } \Lambda_2 \hat{\rho}_m = 0, \tag{3.27}$$

where Λ_1 and Λ_2 are the Lagrangian multipliers of $\hat{\rho}_m$. Derivative of $\mathcal{L}_m(\boldsymbol{\rho}_m, \mathbf{t}_n)$ *w.r.t.* $\hat{\rho}_m$ is given as

$$\frac{d\mathcal{L}_m(\boldsymbol{\rho}_m, \mathbf{t}_n)}{d\hat{\rho}_m} = \hat{t}_n^{\star} + \Delta\hat{t}_n(\hat{\rho}_m - x_m - \Lambda_1) + \Lambda_2, \tag{3.28}$$

where $\Delta\hat{t}_n = \frac{-\delta_n}{2\hat{R}_n^2} + \frac{\delta_n \sum_{c=1}^{C}\sum_{m=1}^{M} 2\hat{R}_n^2}{4\hat{R}_n^4}$. Since $\hat{\rho}_m > 0$, hence from constraint of Equation 3.27 and Equation 3.28, we obtain

$$\hat{\rho}_m^{\star} = \frac{\Delta\hat{t}_n x_m - \hat{t}_n^{\star}}{\Delta\hat{t}_n}. \tag{3.29}$$

Theorem 3.3 *The LG admits a unique optimal best response strategies of the LNs.*

Proof: The second order partial derivative of $U_m(\boldsymbol{\rho}_m, \mathbf{t}_n)$ *w.r.t.* $\hat{\rho}_m$ and $\hat{\rho}_{\bar{m}}$ is given as

$$\begin{bmatrix} H_{11} & 0 & \dots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & H_{MM} \end{bmatrix}, \tag{3.30}$$

where $H_{mm} = \frac{-\delta_n}{2\hat{R}_n^2} + \frac{\delta_n \sum_{c=1}^{C}\sum_{m=1}^{M} 2\hat{R}_n^2}{4\hat{R}_n^4}$. The diagonal elements of Hessian matrix are negative for $\frac{\delta_n}{2\hat{R}_n^2} > \frac{\delta_n \sum_{c=1}^{C}\sum_{m=1}^{M} 2\hat{R}_n^2}{4\hat{R}_n^4}$ and off-diagonal elements are zero. Therefore, the Hessian matrix of $U_m(\boldsymbol{\rho}_m, \mathbf{t}_n)$ is strictly negative definite which implies that **Problem 3** is a standard convex maximization problem and hence proved. □

3.4.4 Stackelberg Equilibrium for solving (n,m,s,c)-TA problem

The LoRa network consists of a SE when LGs estimate the optimal price gain from LNs for maximizing their utility while the LNs select their time duration for data forwarding to maximize their benefits. Algorithm 3.2 explains how to find the SE by using the estimated best response strategies of LNs and optimal strategy of each LG by using Algorithm 3.1. The outcome of Algorithm 3.2 is the solution of the (n,m,s,c)-TA Problem, *i.e.*, the time durations ($\hat{t}_n^{\star}$, where $n \in \mathcal{N}$) of using the allocated SFs by the LNs which satisfy the service requirement of the LNs and maximizes the network revenue.

Algorithm 3.2: Stackelberg Equilibrium in LoRa

Input : Precision threshold ω, ϵ, $\tau \leftarrow 0$, $\hat{\rho}_m[0]$;
Output: Optimal strategy $\hat{t}_n^{\star}$ of n and $\hat{\rho}_m^{\star}$ of m;
do
  $\tau \leftarrow \tau + 1$;
  ▷ Follower game: Each n maximizes its net utility Estimate $\hat{t}_n$ of each $n \in \mathcal{N}$ using Algorithm 3.1;
  ▷ Leader game: Use Equation 3.29 for estimating $\hat{\rho}_m[\tau + 1]$
  $\hat{\rho}_m[\tau + 1] = \hat{\rho}_m[\tau] + \epsilon \bigtriangledown_{U_m} (\hat{\rho}_m[\tau])$;
while $(\|\hat{\rho}_m[\tau + 1] - \hat{\rho}_m[\tau]\| < \omega \hat{\rho}_m[\tau])$;

3.5 Implementation of the solution of (n,m,s,c)-TA

This section proposes distributed and centralized algorithms for implementing the solution of (n,m,s,c)-TA problem.

3.5.1 Distributed algorithms for solving (n,m,s,c)-TA problem

In a distributed solution, each LN and LG run Algorithm 3.1 and Algorithm 3.2, respectively, for solving the (n,m,s,c)-TA problem. Due to the limited resources of the LNs, some of them can not always participate for estimating the time duration. Depending on the participation of the LNs, fully distributed or semi-distributed algorithms are proposed for solving (n,m,s,c)-TA problem. For deciding whether to use fully-distributed or semi-distributed algorithm requires $O(N)$ time complexity.

3.5.1.1 Fully-distributed algorithm

The Fully-Distributed (FD) algorithm runs at each LN to compute its best response strategy based on the pricing strategies of LGs and time duration of other LNs. A LN n broadcasts its best response strategy to other LNs which helps to calculate best response strategies of other LNs. The LNs repeat these steps till not finding the stable state as shown in Algorithm 3.3. To reduce the computation load on LNs, the LNs run Algorithm 3.3 only for the feasible set of SFs and LGs.

3.5.1.2 Semi-distributed algorithm

The Semi-Distributed (SD) algorithm is similar to the FD algorithm except for the participation of the LNs in the network as shown in Algorithm 3.3. Some LNs are presented in the network with low power level due to which they can not perform the computation which require high energy. Therefore, such LNs set, denoted by $\mathcal{N}'$, will not change their strategy at each iteration. Other LNs in the network compute their optimal time duration without the participation of these low power level LNs.

3.5.2 Centralized algorithm for solving (n,m,s,c)-TA problem

In the Centralized (CE) algorithm, a LN or the NS acts as a computation node, denoted by n', for estimating the best response strategies of the LNs. Each LN $n \in \mathcal{N}$ in CE algorithm initially communicates the computation power level ϱ_n to the NS and NS nominates a LN from the among LNs as a computation node n' which has maximum computation power, *i.e.* $\varrho_{n'} \geq \varrho_n$ and $1 \leq n \leq N$. The NS nominates itself as a computation node n' if none of the LN in the network have sufficient computation power level, *i.e.*, $\varrho_n < \varrho_{th}$ where ϱ_{th} is the threshold for minimum required computation

Algorithm 3.3: Distributed algorithm

Input : Precision threshold ω, ϵ, $\tau \leftarrow 0$, $\hat{\rho}_m[0]$;
Output: Optimal strategy $\hat{t}_n^\star$ of n and $\hat{\rho}_m^\star$ of m;

if *Fully-Distributed Algorithm* **then**
  All LNs $n \in \mathcal{N}$ send remaining duty cycle and power level to the NS;
if *Semi-distributed Algorithm* **then**
  Selected LNs $n \in \mathcal{N}'$ send remaining duty cycle and power level to the NS;
NS sends feasible set of SFs $\mathcal{F}_n$ to all n;
do
  $\tau \leftarrow \tau + 1$;
  ▷ Follower game: LN n maximizes its net utility
  for *each LN* n **do**
    for *each feasible SF* $f \in \mathcal{F}_n$ **do**
      Each n estimates $\hat{t}_n$ using Algorithm 3.1;
  ▷ Leader game: Using Equation 3.29 for estimating $\hat{\rho}_m[\tau+1]$
  $\hat{\rho}_m[\tau+1] = \hat{\rho}_m[\tau] + \epsilon \bigtriangledown_{U_m} (\hat{\rho}_m[\tau])$;
while $(\|\hat{\rho}_m[\tau+1] - \hat{\rho}_m[\tau]\| < \omega\hat{\rho}_m[\tau])$;

power. The CE algorithm is called as CEL algorithm and CEN algorithm if n' is NS and a LN, respectively. Algorithm 3.4 shows the steps of the centralized algorithm.

3.6 Results and discussion

This section presents the simulation setup and results by evaluation of the derived expressions.

3.6.1 Simulation setup and overview of results

We validate the analytical models for the performance of the game using Network Simulator-3 [47]. LoRaWAN MAC protocol in Network Simulator-3 supports multiple channels, SFs, LGs, and bi-directional networks with a large number of LNs. We have modified the traffic control model (lorawan-tracing-example.cc). We repeat each experiment 100 times and take the average. Simulations consist of one NS and multiple LGs and LNs randomly deployed in a disc-shaped field with a radius of 10 Kms. All LGs and LNs are configured to use the same 125-kHz BW with 868.100 MHz channel fre-

Algorithm 3.4: Centralized algorithm

Input : Precision threshold ω, ϵ, $\tau \leftarrow 0$, $\hat{\rho}_m[0]$;
Output: Optimal strategy $\hat{t}_n^\star$ of n and $\hat{\rho}_m^\star$ of m;

Each $n \in \mathcal{N}$ communicates ϱ_n to NS;
NS computes $j = \arg\max_i \{\varrho_1, \varrho_2, \cdots, \varrho_i, \cdots, \varrho_n\}$;
if $(\varrho_{\geq} \varrho_{th})$ **then**
  $n' \leftarrow n_j$; ▷ Algorithm is called as **CEN Algorithm**;
else
  $n' \leftarrow$ NS; ▷ Algorithm is called as **CEL Algorithm**;
Each $n \in \mathcal{N}$ communicates information $\mathcal{I}_n$ to n' node;
▷ Run at n' node;
do
  $\tau_1 \leftarrow \tau_1 + 1$;
  ▷ Follower game: Each n maximizes its net utility;
  do
    $Flag \leftarrow 0$;
    $\tau_2 \leftarrow \tau_2 + 1$;
    for $n \leftarrow 1$ **to** N **do**
      ▷ Using Equation 3.18 for estimating $\hat{t}_n[\tau_2 + 1]$;
      $\hat{t}_n[\tau_2 + 1] = \frac{a\hat{R}_n - \hat{R}_n \sum_{j=1, j \neq n}^{N} \hat{R}_j \hat{t}_j[\tau_2] - \delta_n \hat{\rho}_m - \lambda_{n,1}}{2(\hat{R}_n)^2}$;
      if $\left(\left\|\hat{t}_n[\tau_2 + 1] - \hat{t}_n[\tau_2]\right\| > \eta\right)$ **then**
        $Flag \leftarrow 1$;
      $\hat{t}_n[\tau_2] \leftarrow \hat{t}_n[\tau_2 + 1]$;
  while $(Flag == 1)$;
  Estimate $\hat{t}_n$ of each $n \in \mathcal{N}$ using Algorithm 3.1;
  ▷ Leader game: Using Equation 3.29 for estimating $\hat{\rho}_m[\tau_1 + 1]$;
  $\hat{\rho}_m[\tau_1 + 1] = \hat{\rho}_m[\tau_1] + \epsilon \bigtriangledown_{U_m} (\hat{\rho}_m[\tau_1])$;
while $(\|\hat{\rho}_m[\tau_1 + 1] - \hat{\rho}_m[\tau_1]\| < \omega \hat{\rho}_m[\tau_1])$;

quency. The bit rate (bits per second) of the LNs are {5470,3125,1760,980,440,225} for $\{SF_7, SF_8, SF_9, SF_{10}, SF_{11}, SF_{12}\}$, respectively. Two CR configurations (4/5 and 4/7) have been exhaustively evaluated (except for the 20-bit physical header, for which a CR of 4/8 is used). We consider that all LNs have a duty cycle of 1% (which translates into 36 seconds per hour) and follow perfect orthogonality in SFs [5]. The communication delay between LNs and LG are 42.4ms and 3.2ms for uplink and downlink, respectively. We assume the battery capacity of each LN is 2400 mAh. Most of the network parameters are obtained from the datasheet of *LoRaWAN Multitech mDot* [48, 49].

3.6.2 Impact of the proposed solution

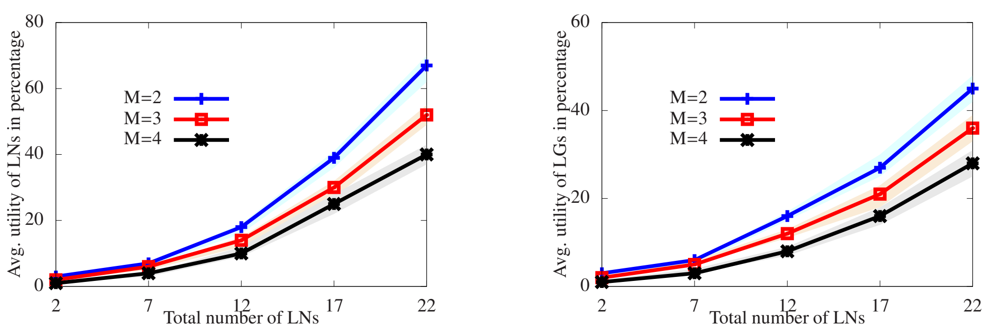
Table 3.1 illustrates the average packet delivery ratio with and without using the proposed solution (Algorithm 3.1 and Algorithm 3.2). Each LN uses the SFs for a fixed time period when the network is not using the proposed solution. It shows that the proposed solution provides high utility as compared with others. This is because if the LG allocates fixed time duration to the LNs for using the SFs then some LNs do not access and over access the network and therefore average utility goes down. It also shows that the difference of the utility with and without using the proposed solution is high in a dense network, *e.g.*, the utilities are 34.6 and 58 for N=22 and M=2. This is because, each LN frees SF as it completes its transmission and other LNs get a chance to transmit data on the free SF which reduces the network congestion or retransmission of data.

Parts (a) and (b) of Figure 3.4 show the results of the total number of LNs versus the average utility in percentage of LNs and LGs, respectively. Here, the average utility of LNs (or LGs) in percentage is calculated as $1/X \sum_{i=1}^{X}(u_a^i - u_u^i)/u_u^i$, where u_a^i and u_u^i are the utility of LNs (or LGs) with and without using the proposed solution, respectively, $1 \leq i \leq X$, and X is the number of LNs (or LGs). Part (a) illustrates that the utility of the LNs increases exponentially as the number of LNs increase in the network. This

Table 3.1: Average packet delivery ratio without and with using the solution.

		Fixed time period			Proposed solution		
		M=2	**M=3**	**M=4**	**M=2**	**M=3**	**M=4**
N	**2**	98.6	98.6	99	98.6	98.8	99.1
	7	94	95.7	96	95.6	96.3	97
	12	82	84	85.7	86	89	91
	17	55.2	61	72.8	77	82	86.5
	22	**34.6**	48	**61**	**58**	67	**76**

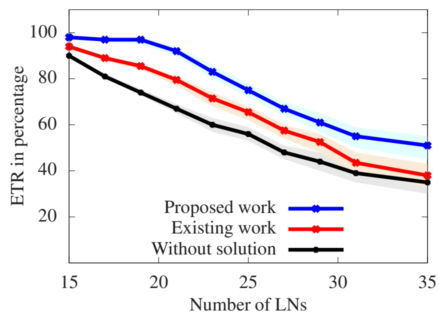
is because, the proposed solution considers the need and requirement of the end users and therefore fully utility the network and increases the gain. Part (b) illustrates that average utility of LGs have less increment as compared with LNs because LGs get revenue for the time allocation whereas LNs get price for data transmission.



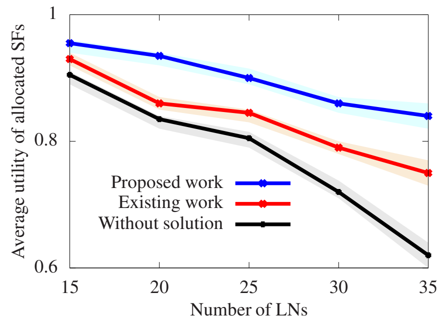
(a) Avg. Utility of LNs. (b) Avg. Utility of LGs.

Figure 3.4: Performance comparison of LNs and LGs between proposed approach and fixed time period SF allocation based approach.

Next, we illustrate the impact of the proposed solution on the ETR and the utility of allocated SFs. The ETR in percentage is the ratio of the achieved TR with interference to the TR without any interference problem. Part (a) of Figure 3.5 shows that if the network uses the proposed solution then interference problem reduces and ETR increases. The exiting work [18] considered the interference problem at the LG and therefore ETR of the exiting work is less than the proposed solution but better than the randomly allocated SFs [16]. The proposed solution considered the need and requirement of the SFs and therefore increases the utility of the allocated SFs as shown in part (b) of Figure 3.5.



(a) Impact of LNs on ETR.



(b) Impact of LNs on utility of SFs.

Figure 3.5: Impact of the proposed solution on ETR and average utility of SFs.

3.6.3 Impact of the parameters of game analysis

This section illustrates the impact of the parameters of Theorem 3.1 (forecasted demand of total data a), Algorithm 3.1 (terminating constant η), and Algorithm 3.2 (terminating constant ω). The experiment considers the LoRa network scenarios where $M = 2$ and $M = 3$ in parts (a1)-(a4) and (b1)-b(4) of Figure 3.6, respectively.

3.6.3.1 Forecasted demand of total data

Parts (a1), (a2), (b1), and (b2) of Figure 3.6 illustrate the impact of the forecasted demand of total data, denoted by a, on the utilities of the LNs and LGs. The results show that the utilities of the LNs and LGs are increasing with the increase of a for the given LNs and LGs, where $a = \{10\%, 15\%, 20\%\}$. This is because, the LNs and LGs forward more data from the end users to the NS when forecasted demand is high. An interesting observation from parts (a1) and (b1) is that, at a given a, the utility of the LNs decreases when the number of LNs increased. This is because, when we increase the LNs in a given network scenario, they get less time duration to access the SFs due to which they are not able to forward the data from end users to the NS and therefore their utilities decrease. Parts (a2) and (b2) illustrate that when the number of LNs increases in the network, the utility of LGs initially increases upto a certain point. This is because, a large number of LNs increases the computation of accessing the limited available SFs at the LGs. However, continuously increasing the LNs creates the congestion in the network and therefore decreases the utility of LGs after a certain point.

3.6.3.2 Terminating constants η and ω

We next present the impact of terminating constant η in Algorithm 3.1 on the utilities of followers and leaders. Parts (a3) and (b3) of Figure 3.6 illustrate that 87 and 98 utility are achieved by the LNs when $M = 2$ and $M = 3$, respectively. It shows that more LGs increases more iterative loop for finding the best SFs for the LNs. Similarly with the parts (a4) and (b4) of Figure 3.6 show the impact of terminating constant ω.

3.6.4 Impact of FD, SD, CEL, and CEN algorithms

The Convergence Rate (CoR) of an algorithm illustrates how much time the algorithm takes for finding the stable state of the network. The CEL and CEN centralized algorithms are running on the NS and a selected LN, respectively. The FD and SD distributed algorithms are running on all LNs and the selected LNs, respectively. The CoR of FD, SD, CEL, and CEN algorithms are denoted by $\Re_{FD}$, $\Re_{SD}$, $\Re_{CEL}$, and $\Re_{CEN}$, respectively. Figure 3.7 illustrates the CoR of algorithms where the LoRa network consists 10 LNs and 2 LGs which are randomly deployed in the FoI. Parts (a) and (b) of Figure 3.7 illustrate the average utility of LNs (ΔU_n) and LGs (ΔU_m) of the given algorithm, respectively, where $\Delta U_n = \sum_{n=1}^{N} U_n(\mathbf{t}_n, \mathbf{t}_{-n})/N$, $\Delta U_m = \sum_{m=1}^{M} U_m(\boldsymbol{\rho}_m, \mathbf{t}_n)/M$, $N = 10$, and $M = 2$. Part (a) shows that the followers (LNs) initially set high time duration for using the allocated SFs. The results illustrate that the followers iteratively update their time duration and the utility eventually converges to a stable value at iteration, confirming the convergence and stability of the algorithms. An interesting

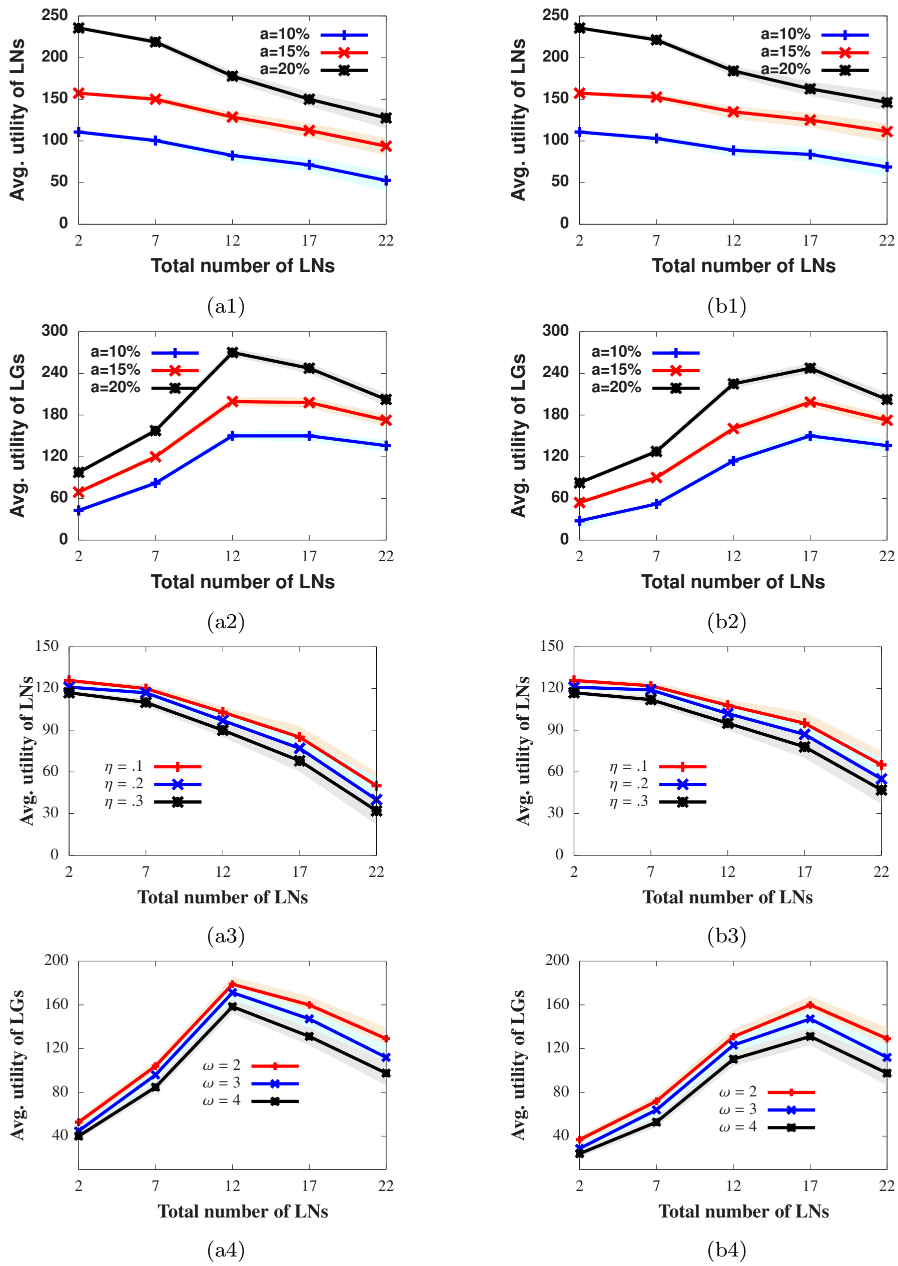
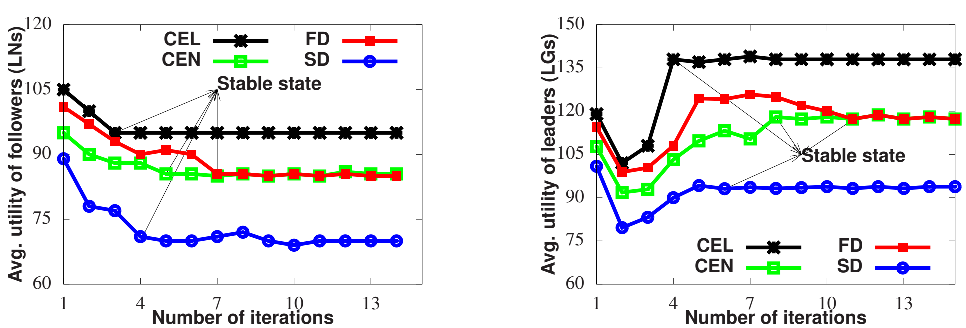


Figure 3.6: Impact of the number of LNs on the utilities of LNs and LGs and rate of convergence. Parts (a1)-(a4) and parts (b1)-(b4) show the results when 2 and 3 LGs deployed in the network, respectively. Parts {(a1),(a2),(a3), and (a4)} show the avg. utility of LNs and LGs for LG=2. Similarly, parts {(b1),(b2),(b3), and (b4)} show the avg. utility of LNs and LGs for LG=3.

observation from this results is the order of CoR is $\Re_{CEL} < \Re_{SD} < \Re_{FD} < \Re_{CEN}$. This is because, the computation power of NS is much higher than the LNs and therefore $\Re_{CEN}$ is highest. The SD algorithm considers limited LNs and therefore quickly finds the stable state than FD algorithm, *i.e.*, $\Re_{FD} > \Re_{SD}$. Finally, CEN algorithm runs on a low processing LN which takes huge time for finding the stable state and therefore CEN is the slowest algorithm. Similarly, the results on the convergence of LGs are shown in part (b) of Figure 3.7. The order of CoR of the algorithms for LGs are the same as LNs. This is because, a LG starts finding its optimal strategy only after the best response strategy of LNs.

Next, we illustrate the impact of algorithms on finding the stable state of the network with different values of the LNs ($N = \{2, 7, 12, 17, 22\}$) and LGs ($M = \{2, 3, 4\}$). Previous result illustrates that CEN algorithm requires maximum time for finding the stable state and therefore consists highest CoR. We calculate the CoR of CEL in percentage of CEN algorithm as $(\Re_{CEN} - \Re_{CEL})\times\ 100/\Re_{CEN}$. Similarly, we calculate the CoR of FD and SD algorithms *w.r.t.* CEN algorithm as shown in Table 3.2. The result illustrates that the CEL algorithm is the fastest algorithm in all algorithms. We therefore conclude that if the network is small then FD is the best algorithm for finding the best solution of (n,m,s,c)-TA problem.



(a) CoR of followers.

(b) CoR of leaders.

Figure 3.7: Illustration of convergence rate of follower (LNs) and leaders (LGs) of FD, SD, CEL, and CEN algorithms.

Table 3.2: CoR in % of CEL, FD, and SD algorithms *w.r.t.* CEN algorithm.

N	$\Re_{CEL}$			$\Re_{SD}$			$\Re_{FD}$		
	No. of LG (M)			**No. of LG (M)**			**No. of LG (M)**		
	2	**3**	**4**	**2**	**3**	**4**	**2**	**3**	**4**
2	1.1	1.3	1.5	1.1	1.3	1.3	1.1	1.3	1.3
7	14.7	19.3	25.3	14.2	18.9	22.6	13.2	17.3	20.4
12	23	30.2	36.7	21.3	28.4	31.3	20.3	25.6	28.4
17	37	40	43	35.3	39.4	41.2	32.1	37.6	38.2
22	45	49	53	40	46	49	38	44	48

3.7 Conclusion

In this chapter, we studied the allocation of SFs based on the needs and requirements of the LNs which helps to handle the interference problem. We estimated the required time of a LN for accessing the SFs such that it satisfies its service requirement and the network maximizes its revenue. Unlike earlier work in the literature, we used the end-to-end network to compute the effective transmission rate and time duration for using the allocated SF with interference problem in the network. We proposed centralized and distributed algorithms to implement the proposed solution. The results demonstrated that the interference among LNs should be considered to compute the transmission rate, particularly for high density of LNs.