Spring 5-13-2023

Incentives to Learn: A Location-based Federated Learning Model

Ryan Kilpatrick-Morrison Brown
University of New Mexico

Ryan Brown

Candidate

Electrical and Computer Engineering

Department

This thesis is approved, and it is acceptable in quality and form for publication:

Approved by the Thesis Committee:

Dr. Eirini Eleni Tsiropoulou, Chair

Dr. Jim Plusquellic, Member

Dr. Iraklis Anagnostopoulos, Member

Incentives to Learn: A Location-based Federated Learning Model

by

Ryan Brown

B.S., University of New Mexico, 2021

THESIS

Submitted in Partial Fulfillment of the
Requirements for the Degree of

Master of Science
Computer Engineering

The University of New Mexico

Albuquerque, New Mexico

May, 2023

Dedication

To Skye

Acknowledgments

I would like to thank my advisor, Professor Eirini-Eleni Tsiropoulou, for all the help and motivation throughout my graduate studies, for providing guidance and pushing me towards success.

Incentives to Learn: A Location-based Federated Learning Model

by

Ryan Brown

B.S., University of New Mexico, 2021

M.S., Computer Engineering, University of New Mexico, 2023

Abstract

Federated Learning (FL) effectiveness depends, among others, on the quality and quantity of the training data and process realized at the end computing nodes. In this paper, we introduce a novel location-based federated learning model, enabled by a low-cost and fast deployable Reconfigurable Intelligent Surfaces (RIS) - based approach that allows to accurately determine the distributed computing nodes positions. Furthermore, in order to train a global model to support different types of smart city applications, while considering two types of servers, offering a prime and common service, respectively, under different costs, the proposed location-based FL model is complemented by an appropriate incentivization mechanism. The latter is based on the theory of Colonel Blotto games and aims at designing the optimal rewards that should be provided to the computing nodes by the servers, in order the former to be properly motivated to exploit a large amount of their raw data towards improving the FL training performance. The outcome of this process depends on the available budget of each server and on each nodes criticality - determined by its position and available data. The performance evaluation of the proposed location-

based federated learning model is obtained via modeling and simulation using a real dataset.

This work has been published in:

Md Sadman Siraj, Md Sahabul Hossain, Ryan Brown, Eirini Eleni Tsiropoulou, and Symeon Papavassiliou, "Incentives to Learn: A Location-based Federated Learning Model," in 2022 Global Information Infrastructure and Networking Symposium (GIIS), page(s) 40-45, 2022 [1].

Contents

List of Figures

Glossary

N	Set of distributed computing nodes where $N = \{1, \ldots, n, \ldots, \lvert N \rvert\}$
S	Set of servers where $S = \{P, C\}$
B^s	Each server's total budget where $\mathbf{b}^s = [b_1^s, \ldots, b_{\lvert N \rvert}^s]$
G	Areas of interest within the smart city where $G = \{1, \ldots, g, \ldots, \lvert G \rvert\}$
L_g	The virtual center of each area of interest where $L_g = (x_g, y_g, z_g)$
L_n	Node coordinates where $L_n = (x_n, y_n, z_n)$
$C_{g,n}$	Node importance factor where $C_{g,n} \in [0, 1]$
D_n	Raw data collected by node
$D_n^{tr.}$	Portion of node raw data used for training
D_n^{test}	Remaining node raw data used for testing accuracy
$\mathbf{W}_n^{s(i)}$	Node local parameters
A_n^s	Accuracy of local parameters
Δt	Clock bias

Chapter 1

Introduction

Federated Learning (FL) was initially introduced by Google in 2016, as a novel technique aiming to train a global model without needing to transfer raw and private data over to a central server. Federated Learning introduces implicit collaboration among distributed computing nodes, which execute machine learning (ML) algorithms on local data and report the local model parameters to the central server. The latter one aggregates the received parameters, updates the global model, reports the updated global parameters to the distributed nodes, and the overall process is repeated iteratively until the global model converges in terms of an acceptable level of accuracy [2]. In this thesis, a location-based Federated Learning model is studied under scenarios of Global Positioning System (GPS) denial by introducing a novel alternative positioning, navigation, and timing approach [3]. The precise knowledge of the distributed nodes' position can be critical for the accuracy of the Federated Learning global model, especially in applications where the data quality is location-dependent, e.g., fire prediction. Also, a novel incentivization mechanism is proposed based on the network economics theory of Colonel Blotto games in order to provide the sufficient rewards to the distributed nodes to participate in the Federated Learning, while simultaneously accounting for the accuracy of the global model.

1.1 Related Work & Motivation

Federated Learning has attracted the interest of the scientific and industrial communities due to its salient characteristics that can support a wide range of real-life applications, ranging from communications to computing to control-based applications. In [4], the authors apply Federated Learning among an Unmanned Aerial Vehicles (UAVs) swarm in order to perform trajectory planning and target recognition. One UAV from the swarm acts as the central server updating the global model, while the rest of the UAVs act as distributed nodes training locally an ML model, based on their own collected data and reporting their local parameters to the leader-UAV. Federated Learning has also been used to predict the energy demand for electric vehicle networks. In [5], an energy demand Federated Learning-based prediction model is proposed, where the charging station provider acts as the central server updating the global energy demand prediction Federated Learning model, by collecting the updated local parameters from the distributed charging stations. Federated Learning has also been used in wireless communications to optimize the beam reflection on reconfigurable intelligent surfaces (RIS) [6–8]. Specifically, a central server collects the users' local parameters and updates the global model in order to predict the RIS elements' optimal phase shifts to optimize the users' achievable data rate. The users' raw data are their channel state information and their experienced data rate. Federated Learning has been applied also in blockchain-based applications to reduce privacy and security concerns related to data sharing. In [9], a bank acts as the central server receiving local parameters from several enterprises interacting with customers in order to recommend financial products to them, while respecting and preserving the customers' privacy-sensitive information.

While the benefits of ML algorithms are becoming widely recognized, the idea of centralizing streams of sensitive data and providing this data to tech companies and/or governments has proven to be very contentious and unpopular. Furthermore, even if all participants are trustworthy, nefarious actors could be monitoring communication links or even modifying the model parameters. So even when only the model parameters are being transmitted for Federated Learning and not the sensitive data, encryption is often used to provide additional privacy protection. The authors in [10] discuss using homomorphic encryption-based algorithm when exchanging data between each Federated Learning participant and the central server. Specifically, the authors proposed a privacy-preserving federated learning algorithm that allowed each user contributing to the global model with their own private key as opposed to [11,12] where users shared the same private key making them vulnerable to insider attacks.

There are a variety of possible applications of Federated Learning in the medical field where HIPAA privacy concerns could preclude the transmission and sharing of patient data; but where machine learning could be utilized to reduce medical diagnostic costs and improve accuracy [13–16]. For example images from X-rays, CT scans, or other radiography could be characterized by Federated Learning algorithms to automatically diagnose broken bones, tumors, coronary artery disease, etc. Another medical application with obvious privacy concerns is drug detection.

It should be noted that Federated Learning is vulnerable to a variety of attacks that can impact model performance and accuracy. One such attack is known as the Data-Poisoning attack. In this attack, one of the participants tampers with the model by creating poor-quality data for training that model which will generate bad parameters. This type of attack can result in high incidence of misclassification. A variant of this attack uses multiple adversarial nodes to boost the effectiveness of this technique [17].

Model Poisoning is another attack which can be more effective than Data Poisoning. In Model Poisoning attacks the adversary modifies or corrupts the updated model which is distributed to all participating nodes. The authors in [18–20] discuss how this is a complicated issue to resolve while also preserving privacy of contributing nodes. In certain applications, the users contributing to the global federated learning model report their local model anonymously making it possible for a malicious contributor to add false models to poison the global. The solution they found included a privacy-preserving gradient to exclude models with low similarity to the global.

Another type of attack that Federated Learning can be subject to is called a Free-Riding Attack. As it sounds, in this attack a node attempts to preserve it's own resources by not participating in the learning process but still benefiting by leeching off the work of the other nodes. In [21–23] they assume the reward for any user contributing to the global model is the accuracy of the global model, in their model of only selfish users the likelihood of free-riding as the number of users increases becomes almost guaranteed.

In many practical Federated Learning use cases, data available at each node can have varying values to the overall accuracy and convergence of the model [24]. So an optimal subset of the nodes might be utilized instead of data from all available nodes. It has been suggested that this optimal subset can be determined utilizing an approach known as Federated Node Selection with Entropy (FedNSE).

Complementary to the developed Federated Learning models that have been developed in the existing literature, the multi-access edge computing technology can support the computing needs of the system and the end-users. In [25], a novel data offloading decision-making framework is proposed, where users have the option to partially offload their data to a complex Multi-access Edge Computing (MEC) environment, consisting of both ground and UAV-mounted MEC servers. Also, the authors in [26] bring artificial intelligence into the UAVs data offloading process in a multi-server Mobile Edge Computing environment, by adopting principles and concepts from game theory and reinforcement learning. Focusing on the profit perspective of the edge computing functionalities, a usage-based pricing policy for allowing the exploitation of the servers computing resources is proposed in [27]. An alternative data offloading framework is discussed in [28] in order to support the energy and time-efficient video processing in surveillance systems based on game theory in satisfaction form. The aspects of Serverless Computing (SC) functions are examined in [29]. Specifically, a flexible resource-sharing paradigm is introduced, to enable the allocation of users' computing tasks in a social cloud computing system offering both Virtual Machines (VMs) and Serverless Computing (SC) functions.

The problem of incentivizing the distributed computing nodes to participate in the Federated Learning process becomes even more challenging, as compared to implementing the training process itself, given that the limited participation of the nodes can deeply impact the training performance. A Stackelberg game is proposed in [30] among the central server (leader) and the distributed nodes (followers) in order to determine the optimal incentives provided by the server and the corresponding computing effort invested by the nodes to train the local ML models. A similar approach is introduced in [31] considering a crowdsourcing platform as the computing nodes, while tackling the problem of communication efficiency during the exchange of local and global model parameters among the participating actors. Similarly, focusing on Federated Learning-based crowdsourcing applications, the authors in [32] aim at identifying fake crowdsourcing tasks to minimize the prediction loss.

The Colonel Blotto game model [33] a commonly adopted game theory framework in problems involving competitive resource allocation. The model involves two players battling over various nodes using their own allocated resources and for each battlefield the player who dedicates the most resources wins the node in a winner take all fashion. The Colonel Blotto game is used in various applications including political and financial competition, cyber physical systems, communication systems, etc. and is valuable being applied alongside a Federated Learning training technique. Federated learning involves collaborating nodes and in real life applications often the nodes need incentives to forfeit a portion of their resources to contribute to the overall global model. The Colonel Blotto game is applied in [34] where they develop a Network Formation game that considers two players with different resource budgets bidding on the edges between various nodes to create a network layer. This paper applies the game very similarly along with federated learning involving two competing servers that bid over local models from various nodes to add to their own global federated learning model. Alternatively Colonel Blotto as used in [35, 36] works for modeling security scenarios where the two players are defined as attacker and defender and the nodes they bid over are represented by data/servers that need protecting.

Despite the novel advances in the Federated Learning field, the problem of determining the computing nodes' position and the impact of not accurately knowing the nodes' position in the Federated Learning training performance has not been yet properly studied and quantified. This problem becomes even more challenging when it is combined with the problem of determining the optimal rewards provided by the central server to the distributed computing nodes in order not only to locally train an ML model, but also to incentivize them to exploit their available data [24].

Over the last few years the need to be able to operate in GPS denied environments has become increasingly important because of an increase in jamming and spoofing incidents [37–40]. Also, certain geographical locations (e.g., urban canyons, the interior of buildings) have inherently poor GPS coverage so it can be undesirable to design systems dependent on GPS availability for full functionality. Signals of opportunity (SOPs) have been discussed as a substitute for GPS or to supplement locational determination in environments with degraded GPS coverage. SOPs are ambient radio signals that are not intended for navigation or timing purposes, such as AM/FM radio, WiFi, cellular, digital television, and low Earth orbit (LEO) satellite signals. Techniques utilizing SOPs to provide Precision Navigation and Timing

(PNT) in GPS-denied environments are discussed in [41].

Fire detection has conventionally used various heat sensors in order to detect nearby fires but because of increased availability and on board processing power in cameras there has been an increase in image fire detection research. Using image fire detection has the advantage over traditional fire detection methods by not requiring human interaction to confirm the existence of a fire. There already exists a number of methods developed for various image classification using Convolutional Neural Networks that have been applied similarly to fire detection. Convolutional Neural Networks rely on assigning weight to aspects of an image and in order to classify without overfitting. The authors in [42] use a modified image edge detection algorithm aimed at accurately and timely detecting flame edges. The algorithm is modified by first adjusting the image contrast in terms of gray level, smoothing the image to eliminate any noise, and then removing any unrelated edges allowing them to successfully track and recognize fire edges. While [43] proposes a fire detection system that limits the processing constraints for each node while still being able to detect a fire within a reasonable amount of time.

1.2 Contributions & Outline

In this thesis, we introduce a novel location-based federated learning model, enabled by a low-cost and fast deployable Reconfigurable Intelligent Surfaces (RIS) - based approach that allows to accurately determine the distributed computing nodes' positions. The nodes' position is of paramount importance regarding the raw data that are available to them in order to locally train the ML models, and can deeply impact several types of Federated Learning-based applications, such as public safety, and surveillance. For example, in a fire safety application, where the Federated Learning model aims at predicting a fire event, the accurate calculation of the participating nodes' positions in specific critical areas, e.g., natural gas infrastructure, hospitals, etc., becomes of high importance as the raw data of those nodes become more impactful in the training of the global model. The nodes' positioning identification becomes even more challenging in cases of GPS-denial, which are common in indoor environments or due to jamming and spoofing of the satellite signals.

The proposed location-based Federated Learning model is further improved by designing the optimal rewards that should be provided to the computing nodes by the central servers, to properly motivate them to exploit a large amount of their raw data towards improving the Federated Learning training performance. To achieve this goal, the theory of Colonel Blotto games is adopted enabling the calculation of the optimal rewards for the servers that aim to train an Federated Learning model. The proposed game-theoretic model jointly incentivizes the computing nodes to learn, i.e., participate in the Federated Learning process, and maximizes the servers' benefit by optimally allocating rewards to the nodes.

The remainder of the thesis is as follows. The system model is presented in Section 2. Section 3 introduces the proposed location-based Federated Learning model, while Section 4 describes the novel incentivization mechanism based on the Colonel Blotto games. A detailed set of numerical results is presented in Section 5 based on a real dataset. Section 6 concludes the thesis.

Chapter 2

System Model

We consider a smart city scenario, where a set of distributed computing nodes $N = \{1, \ldots, n, \ldots, |N|\}$, such as surveillance cameras, users' smartphones, Internet of Things (IoT) nodes, etc., can participate in an FL process to support different applications, such as fire prediction, traffic planning, etc. Two types of central servers are assumed to exist in the smart city supporting and delivering the same application/service, one offering however a prime service (P) with higher accuracy and another one offering a common service (C) with lower accuracy, under different costs. The set of the servers is denoted as $S = \{P, C\}$. Each server $s \in S$ has a total budget B^s to allocate to the computing nodes at each iteration i of the FL process in order to incentivize them to participate in the process, and accordingly invest their computing resources. In the rest of the analysis, the notation of the ith FL iteration is omitted for notation convenience. Let us denote as $\mathbf{b}^s = [b_1^s, \ldots, b_{|N|}^s]$ the server's s budget allocation to the $|N|$ nodes, with $\sum_{n=1}^{|N|} b_n^s = B^s$.

The smart city is virtually divided into areas of interest, where $G = \{1, \ldots, g, \ldots, |G|\}$ denotes their set. The virtual center of each area has coordinates $L_g = (x_g, y_g, z_g)$ and a node belongs to an area based on the minimum distance criterion $n_g = \arg\min_{g \in G}(||L_g - L_n||)$, where $L_n = (x_n, y_n, z_n)$ denotes the node's coordinates. Thus, each node is characterized by a different importance factor $C_{g,n} \in [0,1]$, which in turn represents the importance of the raw data collected from that area. For example, in a fire prediction application, the criticality of a node residing in an area with a natural gas infrastructure is higher compared to a node belonging to a residential area. By combining the node's importance factor $C_{g,n}$ and the amount of raw data D_n that the node collects from the area that it resides, we define the node's criticality $q_n = \frac{C_{g,n} D_n}{\sum_{\forall n \in N} C_{g,n} D_n}$, $q_n \in [0,1]$, which essentially represents the node's potential in contributing to the FL process and accurately training the global models. This information, i.e., $C_{g,n} D_n$, can be "advertised" by the computing nodes to the servers to attract a higher reward, i.e., allocated budget.

Each server s experiences a utility that captures its benefit from collecting the local parameters from the computing nodes by investing its rewards at each iteration of the FL process, and is defined as follows:

$$U^s(\mathbf{b^s}, \mathbf{b^{-s}}, \kappa) = \sum_{n=1}^{|N|} \frac{q_n}{\pi} \arctan\left[\kappa(b_n^s - b_n^{-s})\right] + \frac{q_n}{2} \tag{2.1}$$

where $\mathbf{b^s} = [b_1^s, \dots, b_{|N|}^s], \mathbf{b^{-s}} = [b_1^{-s}, \dots, b_{|N|}^{-s}]$, and $\kappa \in \mathbb{R}^+$ denotes the elasticity factor capturing the fairness in terms of the received utility from the servers. Specifically, small values of κ support a fairness balance among the servers in terms of enjoying some level of utility even if they invested a small amount of rewards (compared to the other server), while still allowing for provisioning of higher utility to the server that invested more rewards. Also, the utility of the other type of server is derived as $U^{-s}(\mathbf{b^s}, \mathbf{b^{-s}}, \kappa) = 1 - U^s(\mathbf{b^s}, \mathbf{b^{-s}}, \kappa)$, given that the total importance of the nodes is finite, i.e., $\sum_{n=1}^{|N|} q_n = 1$ (constant-sum game).

Each node has a total amount of D_n raw data, where a portion $D_n^{tr.}$ is used for training the local model and the rest amount of data D_n^{test} is used to test its accuracy, with $D_n^{tr.} + D_n^{test} = D_n$. The computing nodes are incentivized to train a part $\frac{U^s}{\sum_{\forall s \in S} U^s} D_n^{tr.}$ of their available data based on the announced utility levels by each server and report their local parameters $\mathbf{W}_n^{s(i)}$ to each corresponding server s to update its global FL model. Also, A_n^s denotes the accuracy of the reported local parameters, as determined by the servers via their own testing dataset.

Chapter 3

A Location-based Federated Learning Model

In this section, a novel location-based federated learning model is introduced to account for the importance of the input local parameters to the training of the global model. At the i^{th} iteration of the FL process, the nodes report the outcome of the local training $\mathbf{W}_n^{s(i)}$ to the servers P, C to update the global model. Each server performs the aggregation of the received local parameters $\mathbf{W}_s^{i+1} = \frac{1}{|N|}\sum_{n=1}^{|N|} C_{g,n} \cdot \mathbf{W}_n^{s(i)}$, where $C_{g,n} \in [0, 1]$ denotes the importance of node n based on the area g that the node resides, as explained earlier. The updated global models are broadcasted to the computing nodes in order to be used in the training round of their local model. It is highlighted that our goal is to improve the training outcome of the FL process by accounting for the node's importance level depending on the area that it resides, which is assumed to directly correlate with the quality of the raw data available to the computing nodes.

In order for the proposed location-based FL model to achieve an accurate training outcome, the nodes' location should be accurately determined in order to be categorized in the corresponding areas of interest and derive their raw data importance factor $C_{g,n}$, as described above. However, the nodes' position (x_n, y_n, z_n) is often unknown due to GPS-denial cases, which can be an outcome of several events, such as interference, spoofing, jamming of the satellite signals, or cases of indoor environments.

Several alternative positioning, navigation and timing (PNT) techniques have been introduced in the recent literature, such as passive wide area multilateration, distance measuring equipment, pseudolites, and local systems [44]. However, those techniques suffer by high infrastructure cost, multi-path effects, and clocks' synchronization among the node and the ground infrastructure that transmits at least four signals in order to perform the nodes' positioning. Four singals are needed in order to perform the multilateration technique and determine the node's coordinates (x_n, y_n, z_n), as well as the clock bias Δt among the node and the ground base stations (BS).

In this thesis, we adopt the low-cost and fast deployable technology of Reconfigurable Intelligent Surfaces (RIS) in order to accurately perform the nodes' positioning. RIS can be easily and fast deployed in every surface (static or mobile) and act as a passive reflector of the incoming beams by appropriately constructing the reflecting beam via tuning the phase shifts of the RIS elements [45,46]. This property is adopted to design a reinforcement learning (RL) based PNT solution. Specifically, in a smart city scenario, a plethora of RISs is expected to be developed to support the 6G communications. A node with unknown coordinates can measure the four pseudoranges from the signal transmitted by a BS and the signals reflected on three RISs, as follows: $r_{BS,n} = |\mathbf{r}_n - \mathbf{r}_{BS}| - \Delta t \cdot c$, $r_{BS,R_i,n} = \mathbf{1}_{BS,R_i}(\mathbf{r}_{R_i} - \mathbf{r}_{BS}) + \mathbf{1}_{R_i}(\mathbf{r}_n - \mathbf{r}_{R_i}) - \Delta t \cdot c$, where c denotes the speed of light, $\mathbf{r}_n = (x_n, y_n, z_n)$, $\mathbf{r}_{BS} = (x_{BS}, y_{BS}, z_{BS})$, $\mathbf{r}_{R_i} = (x_{R_i}, y_{R_i}, z_{R_i})$ with $R_i (i = 1, 2, 3)$ denoting a selected set of three RISs by the node to perform its localization. Based on those four measured pseudoranges, the node can determine the unknown variables $x_n, y_n, z_n, \Delta t$ by implementing the Iterative Least Square (ILS) algorithm, as described below:

Step 1: The four pseudorange equations are set equal to zero, and the functions $g_1 - g_4$ are derived. We denote as $\mathbf{x}^{(k)} = (x_n^k, y_n^k, z_n^k, \Delta t^{(k)})$ the vector of the unknown variables.

Step 2: Get the Jacobian (Eq.3.1) and Residual matrices (Eq.3.2).

$$J^{(k)} = \begin{bmatrix} \frac{\partial g_1(\mathbf{x}^{(k)})}{\partial x_n} & \frac{\partial g_1(\mathbf{x}^{(k)})}{\partial y_n} & \frac{\partial g_1(\mathbf{x}^{(k)})}{\partial z_n} & \frac{\partial g_1(\mathbf{x}^{(k)})}{\partial \Delta t} \\ \frac{\partial g_2(\mathbf{x}^{(k)})}{\partial x_n} & \frac{\partial g_2(\mathbf{x}^{(k)})}{\partial y_n} & \frac{\partial g_2(\mathbf{x}^{(k)})}{\partial z_n} & \frac{\partial g_2(\mathbf{x}^{(k)})}{\partial \Delta t} \\ \frac{\partial g_3(\mathbf{x}^{(k)})}{\partial x_n} & \frac{\partial g_3(\mathbf{x}^{(k)})}{\partial y_n} & \frac{\partial g_3(\mathbf{x}^{(k)})}{\partial z_n} & \frac{\partial g_3(\mathbf{x}^{(k)})}{\partial \Delta t} \\ \frac{\partial g_4(\mathbf{x}^{(k)})}{\partial x_n} & \frac{\partial g_4(\mathbf{x}^{(k)})}{\partial y_n} & \frac{\partial g_4(\mathbf{x}^{(k)})}{\partial z_n} & \frac{\partial g_4(\mathbf{x}^{(k)})}{\partial \Delta t} \end{bmatrix} \tag{3.1}$$

$$\mathcal{R}^{(k)} = \begin{bmatrix} g_1(\mathbf{x}^{(k)}) \\ g_2(\mathbf{x}^{(k)}) \\ g_3(\mathbf{x}^{(k)}) \\ g_4(\mathbf{x}^{(k)}) \end{bmatrix} \tag{3.2}$$

Step 3: Determine the least squares problem solution $(\Delta x_n, \Delta y_n, \Delta z_n, \Delta(\Delta t)) = (J^{(k)^T} \cdot J^{(k)})^{-1} \cdot J^{(k)^T} \cdot \mathcal{R}^{(k)}$ and derive the next best estimate $\mathbf{x}^{(k+1)} = (x_n^{(k)} + \Delta x_n, y_n^{(k)} + \Delta y_n, z_n^{(k)} + \Delta z_n, \Delta t^{(k)} + \Delta(\Delta t))$.

The above steps are repeated iteratively until the difference among two sequential iterations is sufficiently small. The ILS algorithm determines the node's position and its accuracy is evaluated based on the geometric dilution of precision parameter $GDOP = \sqrt{\sum_{\forall i} G(i,i)}$, $G = (J^{(k)^T} \cdot J^{(k)})^{-1}$. The lower is the GDOP value, the more accurately the node's position has been determined. The lowest value reported in existing literature is $GDOP = 1.5811$ [47].

It is evident that the accuracy of the nodes' position depends on the selection of the RISs in order to measure the corresponding pseudoranges from the received signals. In this thesis, we introduce a low complexity reinforcement learning (RL) algorithm to enable the nodes, which act as learning agents, to appropriately select three RISs from the plethora of RISs available in the surrounding environment in order to determine their position [48]. We denote as $\mathcal{RIS}$ the set of the available RISs in the examined area, and $\alpha_n = \{R_i, R_j, R_l\}$, $i \neq j \neq l$, $R_i, R_j, R_l \in \mathcal{RIS}$ denotes the node's action of selecting a subset of RISs. The Linear Reward Inaction (LRI) algorithm is adopted to enable the nodes to learn the most beneficial choice of RISs. The probability of selecting the same (Eq.3.3a) or a different action (Eq.3.3b) is derived based on the following probability updating rules [49, 50]:

$$P_r(\alpha_n^{(ite+1)}) = P_r(\alpha_n^{(ite)}) + \beta \cdot r(\alpha_n^{(ite)}) \cdot (1 - P_r(\alpha_n^{(ite)})), \quad \text{if } \alpha_n^{(ite+1)} = \alpha_n^{(ite)} \tag{3.3a}$$

$$P_r(\alpha_n^{(ite+1)}) = P_r(\alpha_n^{(ite)}) - \beta \cdot r(\alpha_n^{(ite)}) \cdot P_r(\alpha_n^{(ite)}), \quad \text{if } \alpha_n^{(ite+1)} \neq \alpha_n^{(ite)} \tag{3.3b}$$

where $\beta \in [0, 1]$ denotes the learning rate, $r(\alpha_n^{(ite)}) = \frac{1.5811}{GDOP(\alpha_n^{(ite)})}$ denotes the learning reward from selecting a strategy $\alpha_n^{(ite)}$, and *ite* denotes the iteration of the LRI algorithm. The LRI algorithm is executed iteratively until the probability of selecting one strategy is close to 1, and it is initiated with $P_r|_{ite=0} = \frac{1}{|\mathcal{RIS}|}$ [51].

Chapter 4

Incentives to Learn: A Generalized Colonel Blotto Game

In this section, we introduce a novel incentivization mechanism based on the theory of Colonel Blotto games to enable the servers to determine the optimal rewards $\mathbf{b^{s*}} = [b_1^{s*}, \ldots, b_{|N|}^{s*}]$, $\forall s \in S$ that should be provided to the nodes in order to incentivize them to process a large amount of their raw data at each iteration i of the FL process. Each node receives a total reward $b_n^* = \sum_{s=1}^{|S|} b_n^{s*}$ and is incentivized to train part of its data $\frac{U^s}{\sum_{\forall s \in S} U^s} D_n^{tr.}$ at the iteration i of the FL process by investing its personal computing resources.

The servers compete among each other to provide appropriate rewards to the nodes in order to ultimately incentivize more the nodes to process their data and contribute to the accuracy improvement of the global FL model of each server. As described in Section 2, each server has a total budget $B^s, \forall s \in S$, that should be sparingly allocated to the nodes, i.e., $\mathbf{b^{s*}} = [b_1^{s*}, \ldots, b_{|N|}^{s*}]$, $\forall s \in S$, with $\sum_{n=1}^{|N|} b_n^s = B^s$. Based on the announced utility of the servers, each node trains the local ML model on $\frac{U^s}{\sum_{\forall s \in S} U^s} D_n^{tr.}$ amount of data, reports its corresponding local parameters $\mathbf{W}_n^{s(i)}$ to each server s. Each server updates its own global model and experiences an accuracy level A_n^s.

The theory of Colonel Blotto (CB) games is adopted to determine the optimal rewards at each iteration i of the FL process, while considering the servers' available budget $B^s, \forall s \in S$ [52]. The traditional CB games consider two types of competitors, who complete among each other over a set of finite battlefields by investing their available resources. The one that invests more resources into a battlefield wins and enjoys the battlefield's value as a benefit, while the one that invests less resources experiences zero benefit, even if it invested its resources. The traditional CB model has been extended into the Generalized Colonel Blotto (GCB) model, where the competitor that loses the battlefield still enjoys a level of benefits given that invested its resources. Under the GCB model, the competitors' utility is captured by a strictly increasing sigmoidal function with respect to the difference of the competitors' invested resources [53].

By adopting the GCB model in our proposed incentivization model, the servers act as the competitors and the computing nodes are equivalent to the battlefields. As defined in Section 2, each node is characterized by its criticality,

$$q_n = \frac{C_{g,n} D_n}{\sum_{\forall n \in N} C_{g,n} D_n}, q_n \in [0,1]$$

which is considered as equivalent to the value of the battlefield. We define the GCB game at each iteration i of the FL process as

$$G = \{S, \{\mathcal{B}^s\}_{\forall s \in S}, \{B^s\}_{\forall s \in S}, N, \{q_n\}_{\forall n \in N}, \{U^s\}_{\forall s \in S}\}$$

where $S = \{P, C\}$ is the set of servers, $\mathcal{B}^s = \{\mathbf{b^s} | \sum_{n=1}^{|N|} b_n^s \leq B^s, b_n^s \geq 0\}$ is the feasible strategy space of server $s \in S$, B^s is the server's available budget in the i^{th} iteration of the FL process in order to provide rewards to the nodes, N is the set of computing nodes, q_n is the criticality level of node n, and U^s is the server's utility function (Eq. 2.1). Each server aims at selfishly optimizing its utility in order to train its FL model, thus, the corresponding minimax problem is formulated.

$$\min_{b^{-s}\in\mathcal{B}^{-s}} \max_{b^{s}\in\mathcal{B}^{s}} U^s(\mathbf{b^s}, \mathbf{b^{-s}}, \kappa) = \sum_{n=1}^{|N|} \frac{q_n}{\pi} \arctan\left[\kappa(b_n^s - b_n^{-s})\right] + \frac{q_n}{2} \tag{4.1}$$

Towards solving the above optimization problem, our goal is to determine a Pure Nash Equilibrium (PNE) that will enable us to derive a stable and optimal allocation of the servers' rewards to the computing nodes.

Theorem 1. *The GCB game* $G = \{S, \{\mathcal{B}^s\}_{\forall s\in S}, \{B^s\}_{\forall s\in S}, N, \{q_n\}_{\forall n\in N}, \{U^s\}_{\forall s\in S}\}$ *has at least one PNE* $\mathbf{b^{C*}} = [b_1^{C*}, \dots, b_{|N|}^{C*}]$ *and* $\mathbf{b^{P*}} = \mathbf{b^{C*}} + [z_1^*, \dots, z_{|N|}^*]$*, where* $\mathbf{b^{C*}} \in \mathcal{B}^C$ *and* $z_{|N|}^*$ *is the positive solution of* $z_{|N|} + \sum_{n=1}^{|N|-1} \sqrt{\frac{1}{\kappa^2 q_{|N|}}\kappa^2 z_{|N|}^2 q_n + q_n - q_{|N|}} = B^P - B^C$ *for* $B^P > B^C$ *and* $z_n^* = \sqrt{(z_{|N|}^*)^2 \frac{q_n}{q_N} + \frac{q_n - q_{|N|}}{q_{|N|}\kappa^2}}, \forall n \in N\backslash\{|N|\}$ *for* $\kappa \geq \max\{\frac{1}{B^P - B^C} \frac{|N|-1}{\sqrt{q_{|N|}(2|N|-1)}}, \frac{1}{B^P-B^C} \sum_{n=1}^{|N|-1} \sqrt{\frac{q_n - q_{|N|}}{q_{|N|}}}\}$.

Proof: Similar steps can be followed as for Theorem 3 in [53]. Due to space limitations, the proof is omitted here. ■

Based on the above analysis, each server can determine the optimal allocated rewards $\mathbf{b^{s*}}, \forall s \in S$ at each iteration i of the FL process in order to optimize its benefit in terms of accurately training its global model.

Chapter 5

Numerical Results

In this section, the performance evaluation of the proposed location-based federated learning model is obtained via modeling and simulation, in order to reveal its benefits and tradeoffs. In particular, Section 5.1 presents the pure operation and the characteristics of the proposed model, while a comparative evaluation against other alternative models -in terms of identifying the computing nodes positions - is provided in Section 5.2. A real dataset of 2400 images containing 802 fire images and 1538 non-fire images has been used throughout the simulation-based evaluation [54]. The proposed framework's evaluation was conducted in a Dell Tower Desktop with Intel i7 11700K 3.60GHz processor, 32 GB available RAM, and an NVIDIA GeForce GTX 1660 Ti with a 6 GB GDDR6 video memory. The computing nodes are training a local end-to-end deep learning model which takes the fire or non-fire images as input and provides the class predicted for each input image as the output. The model has two convolutional layers stacked sequentially with each convolutional layer followed by a max-pooling layer. The high-level features from the convolutional layers are flattened into vectors which are then passed on to two fully connected layers. The first and second convolutional layers have 8 and 16 filters respectively while the first and second fully connected layers have 10 activation units and 1 activation unit respectively. The model uses the Adam optimizer and loss function based on the binary cross entropy between the actual and predicted classes [43]. In the rest of the analysis, $I = 20$ FL training iterations are examined, while considering the following parameters: $|N| = 5$, $B^C = 800$, $B^P = 1000$, $|G| = 5$, $D_n \in [96, 203, 309, 444, 642]$, $\kappa = 10$, $c = 299792458\ m/s$, $|\mathcal{RIS}| = 10$, $\beta = 0.01$, $\mathbf{C_g} = [0.0625, 0.125, 0.25, 0.5, 1.0]$, unless otherwise explicitly stated.

5.1 Pure Operation

Fig. 5.1a - Fig. 5.1d show the accuracy of the server P, the accuracy of the server C, the node's criticality for each computing node within the examined system, and the global accuracy achieved by each server, as a function of the FL iterations, respectively. Moreover, Fig. 5.2a - Fig. 5.2c demonstrate, for every server (P and C) separately, the average percentage of processed data by each node, the average received rewards by each node, and the average server's utility over the total number of FL iterations, as a function of the nodes' ID.

The results reveal that the nodes of higher criticality (Fig. 5.1c) contribute to a higher achieved accuracy for both servers P (Fig. 5.1a) and C (Fig. 5.1b), and thus, the servers provide them with a higher amount of rewards (Fig. 5.2b) in order to further incentivize them to contribute more data to the FL process and exploit their local computing resources to better train the local ML model. It should be highlighted that the nodes' criticality is not constant over all the FL iterations (Fig. 5.1c), given the accuracy of the proposed Iterative Least Square algorithm that determines the nodes' position, as described in Section 3. Furthermore, it is observed that the server offering prime service (i.e., server P), which is characterized by higher budget B^P, results in providing a larger amount of rewards to the computing nodes (Fig. 5.2b) in order to incentivize them to participate in the FL process. In addition,

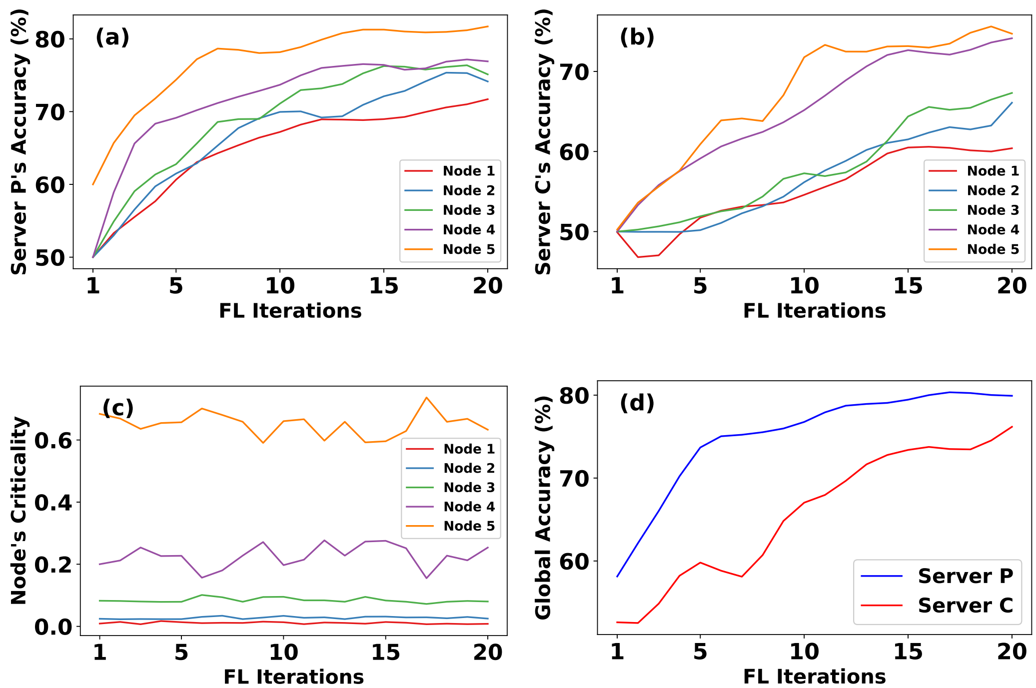

Figure 5.1: Pure operation of the proposed location-based federated learning framework: (a) Accuracy of Server P, (b) Accuracy of Server C, (c) Node's Criticality, and (d) Global Accuracy of FL models as a function of the FL iterations.

the server with a higher available budget aims at incentivizing with higher rewards the nodes with higher criticality regarding their contribution in the improvement of the accuracy of the server's global FL model. The latter observation concludes to a higher portion of data being processed by the more critical nodes for the server that provided a higher reward (Fig. 5.2a). Therefore, the server with higher budget receives higher utility from the more critical nodes, while the server with the lower budget receives a complementary utility (Fig. 5.2c), due to the constant-sum game property, as explained in Section 2.

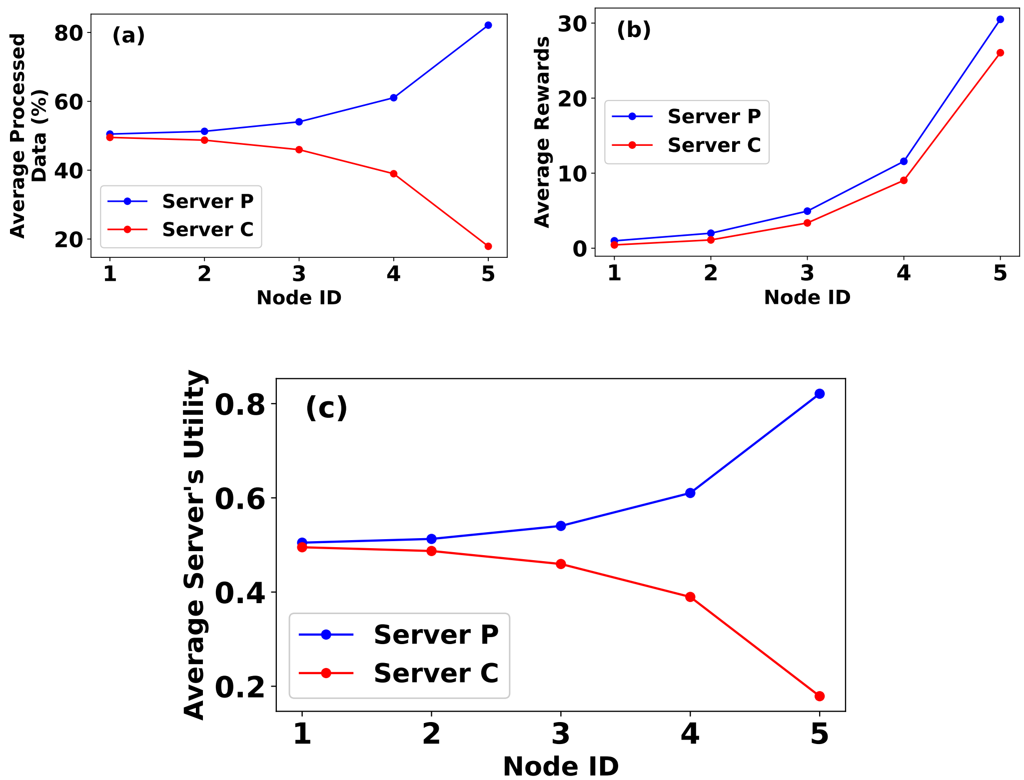

Figure 5.2: Pure operation of the proposed location-based federated learning framework: (a) Average Percentage of Processed Data, (b) Average Rewards, and (c) Average Servers' Utility as a function of the computing nodes' ID.

5.2 Comparative Evaluation

In this section, a detailed comparative evaluation analysis is presented considering four different scenarios regarding the nodes' positioning characteristics, and corresponding knowledge from the servers' point of view. Specifically, the following scenarios are examined: (i) Scenario 1: the servers have perfect knowledge of the nodes' position, (ii) Scenario 2: the nodes' position is determined based on the proposed alternative positioning, navigation, and timing solution presented in this thesis (Section 3), (iii) Scenario 3: the servers have a noisy estimate of the nodes' position,

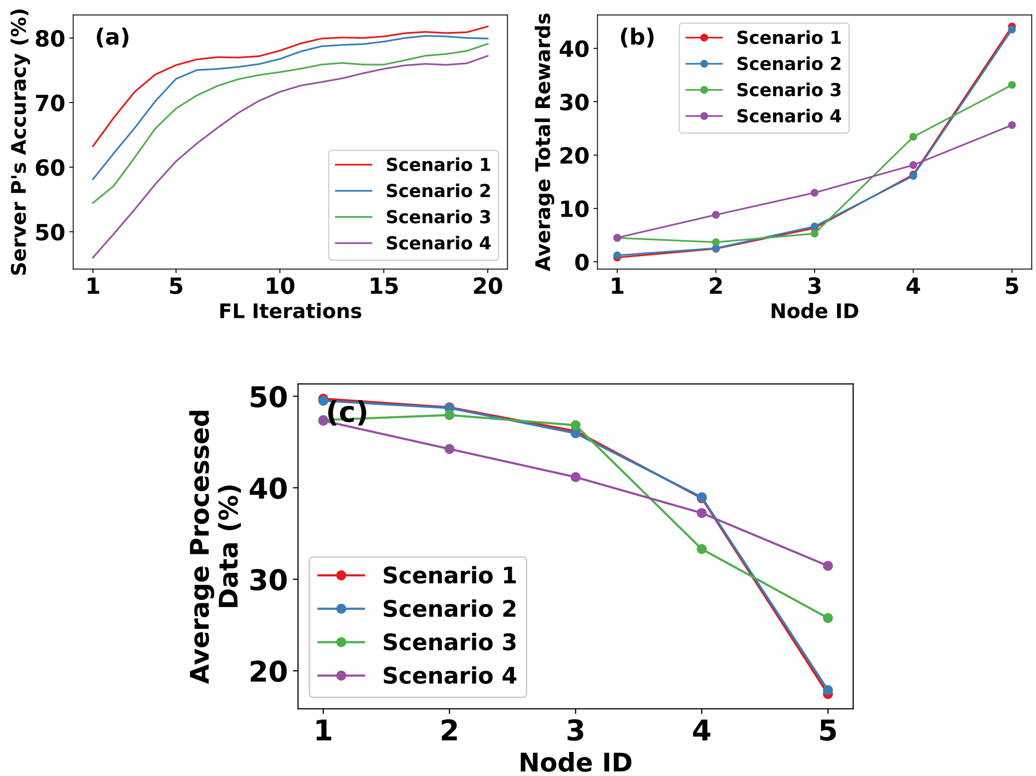

Figure 5.3: Comparative study: (a) Global Accuracy of Server P as a function of the FL iterations, (b) Nodes' Average Total Rewards, and (c) Nodes' Average Percentage of Processed Data.

where the noise is introduced in such a way that a node has equal probability to be erroneously detected at the adjacent regions, and (iv) Scenario 4: the system experiences a GPS denial, and the nodes' position is unknown to the servers. It is highlighted that based on the nodes' position and the corresponding knowledge from the servers' side, the servers can determine the nodes' criticality q_n in terms of contributing in the FL process.

Fig. 5.3a - Fig. 5.3c illustrate the global accuracy of the server P as a function of the FL iterations, the average total rewards, and the average percentage of processed data as a function of the nodes' ID, respectively, considering the four comparative scenarios. The results reveal that the proposed alternative positioning, navigation, and timing (PNT) solution, introduced in this thesis, achieves to accurately determine the nodes' position, thus, resulting in a server's global accuracy of the FL model very close to the one achieved under the scenario of having perfect knowledge of the nodes' position, i.e., only 1.87% less (Fig. 5.3a). Furthermore, it is observed that based on the accurate estimation of the nodes' position, following the proposed alternative PNT solution, the servers provide the same rewards to the nodes as in the (ideal) scenario of assuming perfect knowledge of the nodes' position (Fig. 5.3b), thus, equivalently incentivizing the nodes to process the same average percentage of data (Fig. 5.3c). On the other hand, the scenarios of having a noisy estimation of the nodes' position (Scenario 3), and the scenario of experiencing a GPS denial (Scenario 4) demonstrate the worse accuracy of the global FL model of server P (Fig. 5.3a). This is due to the fact that the servers do not provide their rewards to the nodes in a targeted manner (Fig. 5.3b), which in turn results in not efficiently incentivizing them to process their data (Fig. 5.3c).

Chapter 6

Conclusion

In this thesis, a novel location-based federated learning model is introduced in order to train a global model to support different types of smart cities applications, while considering two types of servers and supporting services, i.e., prime and common. Initially, an alternative low-cost positioning approach is introduced exploiting the RIS technology in order to accurately determine the computing nodes' position, and derive their importance in the FL process based on their access to raw data. Then, a novel incentivization mechanism is introduced based on the theory of Colonel Blotto games, in order to determine the servers' optimal provided rewards to the computing nodes and accordingly mobilize them to determine the level of their participation in the FL process and invest their computing resources to process their data. A detailed numerical evaluation is presented under different scenarios to reveal the benefits and tradeoffs of the proposed location-based federated learning model. Part of our current and future work is the extension of the proposed model in a smart city scenario, where the nodes are characterized by different mobility patterns, and our goal is to develop a mobile crowdsourcing location-based federated learning model.

References

[1] M. S. Siraj, M. S. Hossain, R. Brown, E. E. Tsiropoulou, and S. Papavassiliou, "Incentives to learn: A location-based federated learning model," in *2022 Global Information Infrastructure and Networking Symposium (GIIS)*, 2022, pp. 40–45.

[2] M. Aledhari, R. Razzak, R. M. Parizi, and F. Saeed, "Federated learning: A survey on enabling technologies, protocols, and applications," *IEEE Access*, vol. 8, pp. 140 699–140 725, 2020.

[3] M. S. Siraj, A. B. Rahman, M. Diamanti, E. E. Tsiropoulou, S. Papavassiliou, and J. Plusquellic, "Orchestration of reconfigurable intelligent surfaces for positioning, navigation, and timing," in *MILCOM 2022 - 2022 IEEE Military Communications Conference (MILCOM)*, 2022, pp. 148–153.

[4] T. Zeng, O. Semiari, M. Mozaffari, M. Chen, W. Saad, and M. Bennis, "Federated learning in the sky: Joint power allocation and scheduling with uav swarms," in *IEEE International Conference on Communications (ICC)*, 2020, pp. 1–6.

[5] Y. Saputra, D. Hoang, D. Nguyen, E. Dutkiewicz, M. D. Mueck, and S. Srikanteswara, "Energy demand prediction with federated learning for electric vehicle networks," in *IEEE GLOBECOM*, 2019, pp. 1–6.

[6] D. Ma, L. Li, H. Ren, D. Wang, X. Li, and Z. Han, "Distributed rate optimization for intelligent reflecting surface with federated learning," in *IEEE International Conference on Communications Workshops (ICC Workshops)*, 2020, pp. 1–6.

[7] M. Diamanti, P. Charatsaris, E. E. Tsiropoulou, and S. Papavassiliou, "The prospect of reconfigurable intelligent surfaces in integrated access and backhaul networks," *IEEE Transactions on Green Communications and Networking*, vol. 6, no. 2, pp. 859–872, 2022.

[8] M. S. Hossain, N. Irtija, E. E. Tsiropoulou, J. Plusquellic, and S. Papavassiliou, "Reconfigurable intelligent surfaces enabling positioning, navigation, and timing services," in *ICC 2022 - IEEE International Conference on Communications*, 2022, pp. 4625–4630.

[9] J. Liu, X. He, R. Sun, X. Du, and M. Guizani, "Privacy-preserving data sharing scheme with fl via mpc in financial permissioned blockchain," in *IEEE International Conference on Communications*, 2021, pp. 1–6.

[10] J. Park, N. Y. Yu, and H. Lim, "Privacy-preserving federated learning using homomorphic encryption with different encryption keys," in *2022 13th International Conference on Information and Communication Technology Convergence (ICTC)*, 2022, pp. 1869–1871.

[11] C. Zhang, S. Li, J. Xia, W. Wang, F. Yan, and Y. Liu, "BatchCrypt: Efficient homomorphic encryption for Cross-Silo federated learning," in *2020 USENIX Annual Technical Conference (USENIX ATC 20)*. USENIX Association, Jul. 2020, pp. 493–506. [Online]. Available: https://www.usenix.org/conference/atc20/presentation/zhang-chengliang

[12] Z. Jiang, W. Wang, and Y. Liu, "Flashe: Additively symmetric homomorphic encryption for cross-silo federated learning," *arXiv preprint arXiv:2109.00675*, 2021.

[13] D. Ng, X. Lan, M. M.-S. Yao, W. P. Chan, and M. Feng, "Federated learning: a collaborative effort to achieve better medical imaging models for individual sites that have small labelled datasets," *Quantitative Imaging in Medicine and Surgery*, vol. 11, no. 2, p. 852, 2021.

[14] M. J. Sheller, B. Edwards, G. A. Reina, J. Martin, S. Pati, A. Kotrotsou, M. Milchenko, W. Xu, D. Marcus, R. R. Colen *et al.*, "Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data," *Scientific reports*, vol. 10, no. 1, pp. 1–12, 2020.

[15] E. Darzidehkalani, M. Ghasemi-Rad, and P. van Ooijen, "Federated learning in medical imaging: part i: toward multicentral health care ecosystems," *Journal of the American College of Radiology*, vol. 19, no. 8, pp. 969–974, 2022.

[16] ——, "Federated learning in medical imaging: Part ii: methods, challenges, and considerations," *Journal of the American College of Radiology*, vol. 19, no. 8, pp. 975–982, 2022.

[17] G. Sun, Y. Cong, J. Dong, Q. Wang, L. Lyu, and J. Liu, "Data poisoning attacks on federated machine learning," *IEEE Internet of Things Journal*, vol. 9, no. 13, pp. 11 365–11 375, 2022.

[18] M. Xu, "Feddbg: Privacy-preserving dynamic benchmark gradient in federated learning against poisoning attacks," in *2022 International Conference on Networking and Network Applications (NaNA)*, 2022, pp. 483–488.

[19] C. Richards, S. Khemani, and F. Li, "Evaluation of various defense techniques against targeted poisoning attacks in federated learning," in *2022 IEEE 19th International Conference on Mobile Ad Hoc and Smart Systems (MASS)*, 2022, pp. 693–698.

[20] L. Shi, Z. Chen, Y. Shi, G. Zhao, L. Wei, Y. Tao, and Y. Gao, "Data poisoning attacks on federated learning by using adversarial samples," in *2022 International Conference on Computer Engineering and Artificial Intelligence (ICCEAI)*, 2022, pp. 158–162.

[21] Y. E. Sagduyu, "Free-rider games for federated learning with selfish clients in nextg wireless networks," in *2022 IEEE Conference on Communications and Network Security (CNS)*, 2022, pp. 365–370.

[22] Y. Fraboni, R. Vidal, and M. Lorenzi, "Free-rider attacks on model aggregation in federated learning," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2021, pp. 1846–1854.

[23] J. Wang, X. Chang, R. J. Rodrìguez, and Y. Wang, "Assessing anonymous and selfish free-rider attacks in federated learning," in *2022 IEEE Symposium on Computers and Communications (ISCC)*, 2022, pp. 1–6.

[24] S. Bansal, M. Bansal, R. Verma, R. Shorey, and H. Saran, "Fednse: Optimal node selection for federated learning with non-iid data," in *2023 15th International Conference on COMmunication Systems & NETworkS (COMSNETS)*, 2023, pp. 713–721.

[25] P. A. Apostolopoulos, G. Fragkos, E. E. Tsiropoulou, and S. Papavassiliou, "Data offloading in uav-assisted multi-access edge computing systems under resource uncertainty," *IEEE Transactions on Mobile Computing*, vol. 22, no. 1, pp. 175–190, 2021.

[26] G. Fragkos, N. Kemp, E. E. Tsiropoulou, and S. Papavassiliou, "Artificial intelligence empowered uavs data offloading in mobile edge computing," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, 2020, pp. 1–7.

[27] G. Mitsis, E. E. Tsiropoulou, and S. Papavassiliou, "Data offloading in uav-assisted multi-access edge computing systems: A resource-based pricing and user risk-awareness approach," *Sensors*, vol. 20, no. 8, p. 2434, 2020.

[28] P. A. Apostolopoulos, M. Torres, and E. E. Tsiropoulou, "Satisfaction-aware data offloading in surveillance systems," in *Proceedings of the 14th workshop on challenged networks*, 2019, pp. 21–26.

[29] P. A. Apostolopoulos, E. E. Tsiropoulou, and S. Papavassiliou, "Risk-aware social cloud computing based on serverless computing model," in *2019 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2019, pp. 1–6.

[30] Y. Zhao, Z. Liu, C. Qiu, X. Wang, F. R. Yu, and V. C. Leung, "An incentive mechanism for big data trading in end-edge-cloud hierarchical federated learning," in *IEEE GLOBECOM*, 2021, pp. 1–6.

[31] S. R. Pandey, N. H. Tran, M. Bennis, Y. K. Tun, Z. Han, and C. S. Hong, "Incentivize to build: A crowdsourcing framework for federated learning," in *IEEE GLOBECOM*, 2019, pp. 1–6.

[32] Z. Chen, M. Simsek, and B. Kantarci, "Federated learning-based risk-aware decision to mitigate fake task impacts on crowdsensing platforms," in *IEEE International Conference on Communications*, 2021, pp. 1–6.

[33] B. Roberson, "The colonel blotto game," *Economic Theory*, vol. 29, no. 1, pp. 1–24, 2006.

[34] W. Saad, Z. Han, M. Debbah, and A. Hjorungnes, "A distributed coalition formation framework for fair user cooperation in wireless networks," *IEEE Transactions on Wireless Communications*, vol. 8, no. 9, pp. 4580–4593, 2009.

[35] M. Min, L. Xiao, C. Xie, M. Hajimirsadeghi, and N. B. Mandayam, "Defense against advanced persistent threats: A colonel blotto game approach," in *2017 IEEE International Conference on Communications (ICC)*, 2017, pp. 1–6.

[36] L. Zhang, Y. Wang, M. Min, C. Guo, V. Sharma, and Z. Han, "Privacy-aware laser wireless power transfer for aerial multi-access edge computing: A colonel blotto game approach," *IEEE Internet of Things Journal*, pp. 1–1, 2022.

[37] D. Borio, F. Dovis, H. Kuusniemi, and L. Lo Presti, "Impact and detection of gnss jammers on consumer grade satellite navigation receivers," *Proceedings of the IEEE*, vol. 104, no. 6, pp. 1233–1245, 2016.

[38] M. L. Psiaki and T. E. Humphreys, "Gnss spoofing and detection," *Proceedings of the IEEE*, vol. 104, no. 6, pp. 1258–1270, 2016.

[39] C. J. Hegarty, D. Bobyn, J. Grabowski, and A. Van Dierendonck, "An overview of the effects of out-of-band interference on gnss receivers," in *Proceedings of the 24th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2011)*, 2011, pp. 1941–1956.

[40] R. T. Ioannides, T. Pany, and G. Gibbons, "Known vulnerabilities of global navigation satellite systems, status, and potential mitigation techniques," *Proceedings of the IEEE*, vol. 104, no. 6, pp. 1174–1194, 2016.

[41] Z. M. Kassas, J. Khalife, A. A. Abdallah, and C. Lee, "I am not afraid of the gps jammer: Resilient navigation via signals of opportunity in gps-denied environments," *IEEE Aerospace and Electronic Systems Magazine*, vol. 37, no. 7, pp. 4–19, 2022.

[42] T. Qiu, Y. Yan, and G. Lu, "An autoadaptive edge-detection algorithm for flame and fire image processing," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 5, pp. 1486–1493, 2012.

[43] K. Muhammad, J. Ahmad, Z. Lv, P. Bellavista, P. Yang, and S. W. Baik, "Efficient deep cnn-based fire detection and localization in video surveillance applications," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 7, pp. 1419–1434, 2019.

[44] S. Han, Z. Gong, W. Meng, C. Li, and X. Gu, "Future alternative positioning, navigation, and timing techniques: A survey," *IEEE Wireless Communications*, vol. 23, no. 6, pp. 154–160, 2016.

[45] M. Diamanti, E. E. Tsiropoulou, and S. Papavassiliou, "The joint power of noma and reconfigurable intelligent surfaces in swipt networks," in *2021 IEEE 22nd International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2021, pp. 621–625.

[46] M. Diamanti, M. Tsampazi, E. E. Tsiropoulou, and S. Papavassiliou, "Energy efficient multi-user communications aided by reconfigurable intelligent surfaces and uavs," in *2021 IEEE International Conference on Smart Computing (SMARTCOMP)*, 2021, pp. 371–376.

[47] M. Zhang and J. Zhang, "A fast satellite selection algorithm: beyond four satellites," *IEEE Journal of Selected Topics in Signal Processing*, vol. 3, no. 5, pp. 740–747, 2009.

[48] G. Fragkos, C. Minwalla, J. Plusquellic, and E. E. Tsiropoulou, "Artificially intelligent electronic money," *IEEE Consumer Electronics Magazine*, vol. 10, no. 4, pp. 81–89, 2020.

[49] G. Fragkos, P. A. Apostolopoulos, and E. E. Tsiropoulou, "Escape: Evacuation strategy through clustering and autonomous operation in public safety systems," *Future Internet*, vol. 11, no. 1, p. 20, 2019.

[50] G. Fragkos, E. E. Tsiropoulou, and S. Papavassiliou, "Disaster management and information transmission decision-making in public safety systems," in *2019 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2019, pp. 1–6.

[51] M. Diamanti, G. Fragkos, E. E. Tsiropoulou, and S. Papavassiliou, "Unified user association and contract-theoretic resource orchestration in noma heterogeneous wireless networks," *IEEE Open Journal of the Communications Society*, vol. 1, pp. 1485–1502, 2020.

[52] A. B. Rahman, M. S. Siraj, N. Kubiak, E. E. Tsiropoulou, and S. Papavassiliou, "Network economics-based crowdsourcing in online social networks," in *GLOBECOM 2022-2022 IEEE Global Communications Conference*. IEEE, 2022, pp. 4655–4660.

[53] A. Ferdowsi, A. Sanjab, W. Saad, and T. Basar, "Generalized colonel blotto game," in *Annual Am. Control Conf.* IEEE, 2018, pp. 5744–5749.

[54] "Fire detection dataset," https://www.kaggle.com/datasets/christofel04/fire-detection-dataset.