Incentive-Driven Federated Learning and Associated Security Challenges: A Systematic Review

SUBMISSION DATE / POSTED DATE

10-07-2021 / 14-07-2021

CITATION

Ali, Asad; Ilahi, Inaam; Qayyum, Adnan; Mohammed, Ihab; Al-Fuqaha, Ala; Qadir, Junaid (2021): Incentive-Driven Federated Learning and Associated Security Challenges: A Systematic Review. TechRxiv. Preprint. https://doi.org/10.36227/techrxiv.14945433.v1

DOI

10.36227/techrxiv.14945433.v1

Incentive-Driven Federated Learning and Associated Security Challenges: A Systematic Review

Asad Ali[1], Inaam Ilahi[1], Adnan Qayyum[1], Ihab Mohammed[2], Ala Al-Fuqaha[3], and Junaid Qadir[1]
[1] Information Technology University (ITU), Punjab, Lahore, Pakistan
[2] Western Illinois University (WIU), Illinois, USA
[3] Hamad Bin Khalifa University (HBKU), Doha, Qatar

Abstract

In response to various privacy risks, researchers and practitioners have been exploring different paradigms that can leverage the increased computational capabilities of consumer devices to train machine (ML) learning models in a distributed fashion without requiring the uploading of the training data from individual devices to central facilities. For this purpose, federated learning (FL) was proposed as a technique that can learn a global machine model at a central master node by the aggregation of models trained locally using private data. However, organizations may be reluctant to train models locally and to share these local ML models due to required computational resources for model training at their end and due to privacy risks that may result from adversaries inverting these models to infer information about the private training data. Incentive mechanisms have been proposed to motivate end users to participate in collaborative training of ML models (using their local data) in return for certain rewards. However, the design of an optimal incentive mechanism for FL is challenging due to its distributed nature and the fact that the central server has no access to clients' hyperparameters information and the amount/quality data used for training, which makes the task of determining the reward based on the contribution of individual clients in FL environment difficult. Even though several incentive mechanisms have been proposed for FL, a thorough up-to-date systematic review is missing and this paper fills this gap. According to the best of our knowledge, this paper is the first systematic review that comprehensively enlists the design principles required for implementing these incentive mechanisms and then categorizes various incentive mechanisms according to their design principles. In addition, we also provide a comprehensive overview of security challenges associated with incentive-driven FL. Finally, we highlight the limitations and pitfalls of these incentive schemes and elaborate upon open-research issues that required further research attention.

Index Terms

Federated Learning, Incentive schemes, Incentivization, Game Theory, Mechanism Design, Blockchain

I. Introduction

Recent years have witnessed an upsurge in the use of federated learning (FL) to enhance the privacy of machine learning (ML) applications [1], [2]. FL, also known as collaborative learning, is a technique for training ML models across multiple devices or servers with distributed data samples without sharing the actual data. Particularly, this is useful in applications where the data cannot be shared due to security and privacy constraints, e.g., healthcare. The success of Google's text prediction keyboard [3] along with increased data privacy has opened many new directions for FL, e.g., digital healthcare [4], [5], internet of things (IoT) [6], unmanned vehicles [7], and Industry 4.0 [8].

Due to privacy and security issues, uploading consumers' data to a central cloud may not be deemed practical since the algorithms that involve the use of centralized collections of data pose a great threat to data privacy and integrity [9]. On the other hand, FL helps to alleviate this problem by only sharing the ML model parameters (trained on distributed private data) with a central server, without having to share the users' private data, which contributes to enhanced data privacy. In a simple FL system, an ML model with random weights is uploaded to the server which acts as a master node. In the first step, each end device (ED) (client, user, or data owner)[1] downloads an initial model from the master node and trains the model locally using its computational resources, local data, and hyperparameters. Subsequently, the parameters of trained local models are shared with the model owner (MO) (task publisher or server)[2], where a global model is generated by aggregating the local models' parameters. This whole process is repeated until the desired accuracy is achieved or the objective is fulfilled. We refer interested readers to a few recent comprehensive studies on the fundamentals of FL [10]–[13].

Mobile phones, smart wearable devices, automated vehicles, and smart sensors are a few of the devices that generate a lot of data for model training. Due to their increasing computing capacity and limited transmission capability, it is highly desirable to train ML models locally on these devices. This makes FL extremely useful but its use is limited due to issues such as: (i) expensive communication of models; (ii) heterogeneity of EDs; (iii) statistical diversity of data; (iv) potential revelation of sensitive information from model updates to the common third-party server; (v) costly local training at the EDs; (vi) susceptibility to security threats due to its distributed nature and use of black-box models; and (vii) reluctance of data

[1] Note that the terms data owners, users, client, and end devices are similar and are used interchangeably throughout the paper.

[2] Note that the terms server, master node, publisher, and model owner are used interchangeably throughout the paper.

TABLE I: Comparison with existing survey and review articles of incentive driven FL. Legend: (ID: Incentive-driven, FL: Federated Learning, DRL: Deep Reinforcement Learning, CT: Contract Theory, GT: Game Theory, AT: Auction Theory, BC: Blockchain, MD: Mechanism Design)

Year	Author	Type	Highlights	Incentive Mechanisms						Security Challenges	Limitations	Open Issues
				CT	GT	AT	BC	MD	DRL			
2021	Zhan et al. [16]	Tutorial	- Describe general ID challenges for FL - Presents a case study using DRL for ID-FL	✗	✗	✗	✗	✓	✓	✗	✗	✗
2021	Zhan et al. [17]	Survey	Presents a survey on client-focused ID-FL based on: – Client's data contribution (data quality and quantity) – Client's reputation – Client's resource allocation	✗	✗	✗	✗	✗	✗	✗	✗	✓
2021	Our Paper	Systematic Survey	- Presents systematic review on ID-FL - Develops a pipeline of ID-FL and highlights associated security challenges	✓	✓	✓	✓	✓	✓	✓	✓	✓

TABLE II: Comparison with existing survey and review articles that are generally focused on FL.

Year	Author	Type	Highlights	Incentive	Security	Privacy	Open Issues
2019	Li et al. [18]	Survey	- Provides a thorough categorization of FL systems - Review of infrastructure of FL systems - Describe the design factors for successful FL systems	✗	✗	✓	✓
2019	Kairouz et al. [13]	Survey	- An extensive discussion on problems and challenges in FL	✗	✓	✓	✗
2019	Yang et al. [1]	Survey	- Provides a detailed study on architectures and applications for FL - Comparison of FL with other similar paradigms	✗	✗	✓	✗
2020	Li et al. [10]	Tutorial	- Describe the major challenges for FL - Providing insights to tackles these challenges	✗	✗	✗	✓
2020	Liu et al. [12]	Systematic Review	- Overall research and application trend of FL - Systematically analyzes the approaches to improve the quality of FL models - Compares FL and non-FL algorithms in terms of learning quality	✓	✗	✓	✗
2020	Rahman et al. [11]	Survey	- Compare different ML architectures in context of FL - Classification of FL topics on the basis of technical challenges - A thorough discussion on FL system models, application areas and privacy	✓	✓	✓	✓
2020	Aledhari et al. [19]	Survey	- Provides a comprehensive study on enabling software and hardware platforms and protocols - Provides an overview of key challenges and advantages of FL	✗	✓	✓	✗
2021	Ours	Systematic Review	- Presents systematic review of incentive-driven FL - Develops a pipeline of incentive driven FL and highlights associated security challenges	✓	✓	✗	✓

owners to share their data owing to privacy risks [14]. These issues can make EDs reluctant to participate in the collaborative training process. It becomes highly desirable therefore to provide incentives for data owners to motivate them to actively participate in the FL process [15].

A general principle to incentivize FL is: *"the better the updates provided by the participant; more will be the incentive"* [20]. This principle has been implemented in practical settings for multiple purposes. Similar schemes are employed by organizations to motivate employees so that they perform their level best, e.g., an additional incentive is given to employees who give extra time to a project, or a bonus is given to those whose performance is extraordinary throughout the month. In the same way, incentive schemes have been proposed for FL such that both the user and the publisher of the task benefit from it [15].

Moreover, due to their distributed nature, FL systems are highly vulnerable to malicious attacks that can limit their application in some practical scenarios. FL is susceptible to certain attacks such as data poisoning attacks in which the attacker directly manipulates the training data and sends the malicious updates to the central server or the attacker can directly manipulate the local model. Another such attack is adversarial attacks, in which the attacker tries to fool the ML model by supplying deceiving input. Since the central server does not have access to the user's private data, the users can cheat the server by sending malicious updates or perform adversarial attacks. Therefore, to improve the robustness of real-world FL systems, it is censorious to evaluate how the FL system behaves under such attacks. The risk of such kind of attacks increases further when FL systems are incentivized.

Contributions of the Paper: Significant progress has been made related to incentive mechanisms in FL over the recent years. In this paper, we provide a systematic review of the different incentive mechanisms proposed for FL. The following are the main contributions of this paper:

1) We conduct a systematic review of 46 papers related to incentive mechanisms in FL and the related security challenges.
2) We highlight six approaches used to design incentive mechanisms.
3) We provide a comparison of proposed incentive mechanisms including the discussion of their efficacy.
4) We highlight open research issues and provide insights on current limitations and future research studies.

A comparison of this article with similar articles has been provided in Table I and II.

Organization of Paper: This paper comprises eight sections. Section II discusses the background settings used to devise incentive schemes for FL. Section III provides a systematic review of the paper. Section IV presents the existing work that is being done related to this field. Section V highlights the security threats to FL and the existing solutions to such threats

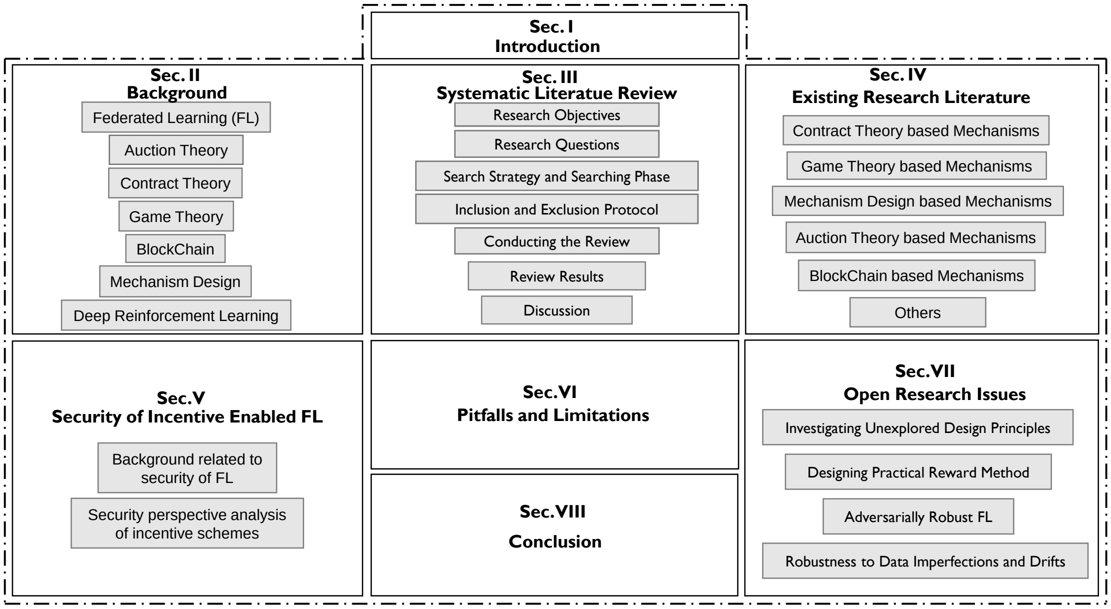
Sec. I
Introduction
Sec. II
Background
Federated Learning (FL)
Auction Theory
Contract Theory
Game Theory
BlockChain
Mechanism Design
Deep Reinforcement Learning
Sec. III
Systematic Literatue Review
Research Objectives
Research Questions
Search Strategy and Searching Phase
Inclusion and Exclusion Protocol
Conducting the Review
Review Results
Discussion
Sec. IV
Existing Research Literature
Contract Theory based Mechanisms
Game Theory based Mechanisms
Mechanism Design based Mechanisms
Auction Theory based Mechanisms
BlockChain based Mechanisms
Others
Sec.V
Security of Incentive Enabled FL
Background related to security of FL
Security perspective analysis of incentive schemes
Sec.VI
Pitfalls and Limitations
Sec.VIII
Conclusion
Sec.VII
Open Research Issues
Investigating Unexplored Design Principles
Designing Practical Reward Method
Adversarially Robust FL
Robustness to Data Imperfections and Drifts

Fig. 1: An overview of the article.

TABLE III: List of acronyms.

Acronym	Expanded Form
AoI	Age of Information
CCM	Committee Consensus Mechanism
CI	Contribution Index
DL	Deep Learning
DNN	Deep Neural Network
DRL	Deep Reinforcement Learning
ED	End Device
FL	Federated Learning
FLI	Federated Learning Incentivizer
GNN	Graph Neural Networks
IID	Independently and Identically Distributed
IIoT	Industrial Internet of Things
IoT	Internet of Thing
IoV	Internet of Vehicles
IPFS	Inter-Planetary File System
MEC	Mobile Edge Computing
MDC	Mobile Device Cloud
ML	Machine Learning
MO	Model Owner
P2P	Peer to Peer
RL	Reinforcement Learning
RMA	Reverse Multi-dimensional Auction
SV	Shapley Value
UAV	Unmanned Aerial Vehicle
VCG	Vickery Clarke Grove

especially in the environment of incentivization. The discussion on all the incentive schemes, their advantages, disadvantages, limitations, and pitfalls are being carried out in Section VI followed by future directions, and open research issues in Section VII. The paper is then concluded in Section VIII. The overview of the paper is provided in Figure 1. The list of different acronyms used in this paper is provided in Table III.

II. Background

In this section, we provide background on FL and the techniques that are used to build incentive mechanisms for FL.

A. Federated Learning

Ensuring data privacy has been a great challenge in traditional ML and Deep Learning (DL) model development. As a solution, FL was proposed by Yang et al. [1] as a step towards ensuring data privacy. FL, also sometimes referred to as on-site ML, is a technique that involves the training of a shared model locally on each user's machine without sharing the private data it owns. Trained ML model parameters are uploaded to the server where they are aggregated to generate a global model. A general architecture of FL containing multiple EDs and a master server is shown in Figure 2.

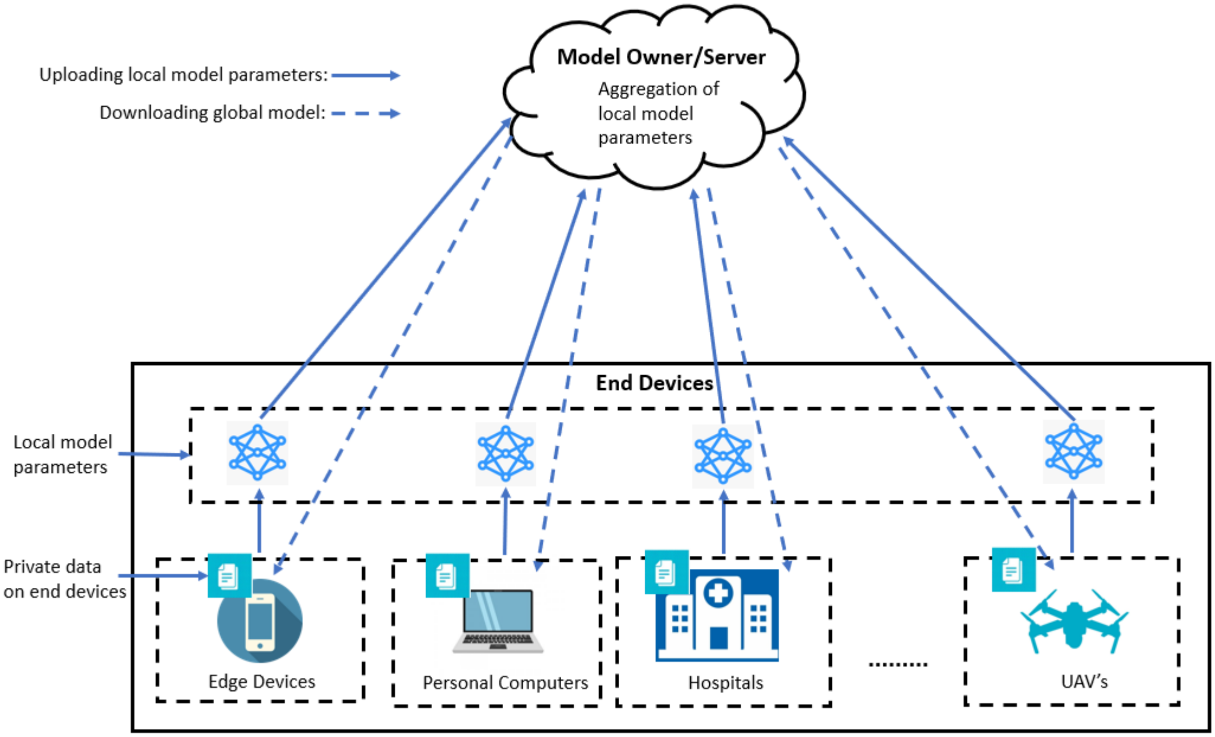
Uploading local model parameters:
Downloading global model:
Model Owner/Server
Aggregation of local model parameters
End Devices
Local model parameters
Private data on end devices
Edge Devices
Personal Computers
Hospitals
....
UAV's

Fig. 2: An illustration of the FL process.

Multiple architectures have been proposed for FL and have been classified by Aledhari et al. [19] based on the benefits and focus of each technique. We do not discuss such schemes in detail as they are beyond the scope of this paper, which is specifically focused on incentive-aware FL.

Mathematically, in FL at t^{th} communication round, a set S_t of n_t data owners download the current model W_t from the sever. Each data owner i locally trains this downloaded model using their data $[D_1, D_2, D_3, \cdots, D_n]$. These local models are denoted by $[W_t^1, W_t^2, W_t^3, \cdots, W_t^n]$, where the update of the ith data owner is given in Eq. 1. These local models are uploaded to the server where they are aggregated using Eq. 2. This aggregation is done using a certain aggregation rule, i.e, it can be FedAvg, secure aggregation protocol [9], Multi-krum [21], Bulyan [22], trimmed mean [23] or median [23]. This aggregated update is used to update the global model using Eq. 3 based on a learning rate η_t [24]. The learning rate is an important hyper-parameter that manages the modification in models in response to error rate provided by testing the updated model using the validation dataset.

$$H_t^i = W_t^i - W_t \tag{1}$$

$$H_t = \frac{1}{n_t}\sum_{i}^{n_t} H_t^i \tag{2}$$

$$W_{t+1} = W_t + \eta_t H_t \tag{3}$$

1) Information Asymmetry: Different levels of information about data owners might be available to the server, i.e., the server might know the types of users (termed as weakly incomplete information) or the server might only know the user type distribution (termed as strongly incomplete information). These different levels of information asymmetry require different strategies to achieve the highest possible model accuracy while keeping the cost low.

2) Types of FL: FL has been divided into three types based on the used data features [25]:

- *Horizontal FL* is used in such settings where two datasets at different regions share overlapping feature space but differ in sample space. For example, consider two banks in different regions that have a small number of overlapping clients. Their business is similar hence the feature space is the same.
- *Vertical FL* is used in scenarios where two datasets share the same sample space but differ in feature space. For example, consider a bank and a hospital in the same region. They might have a large number of overlapping clients but their feature space is different.
- *Federated transfer learning* is used where two datasets differ not only in the sample space but also in the feature space. For example, consider a bank and a hospital in different regions of the world. They may have a small intersection of features and sample space.

B. Contract Theory

Contract theory has been used in various fields such as finance, economics, corporate law, and management [26]. In economics, it is used to design a contract between an employer and an employee. Usually, the employer is not aware of the characteristics of the employee. This situation in contract theory is called "information asymmetry". Contract theory models can tackle information asymmetry by designing contracts with a reward in return for a required performance hence motivating the employees in accepting the contract. The design of a good incentive mechanism plays a vital role in contract theory. This incentive is based on the relationship between the employees' compensation and their performance. The goal in contract theory is to maximize the employees' rewards subject to the incentive constrain of signing the contract such that the rewards are equal or higher than in the case of not signing the contract [27], [28].

Several models have been proposed in contract theory and are categorized as follows [27]:

- **Number of Participants:** Based on the number of participants, a contract theory models can be of two types: (i) *bi-lateral contract* which is a one-to-one contract with one employer and one employee, and (ii) *multi-lateral contract* which is a one-to-many contract with one employer and many employees.
- **Dimensionality of Characteristics:** Based on the the dimensionality of characteristics considered in the contract, a contract theory model can be of two types: (i) *one-dimensional contract* which involves the consideration of only one characteristic, and (ii) *multi-dimensional contract* which involves the consideration of many characteristics.
- **Type:** There could be two types of contract: (i) *static contract* which is a one-shot contract in which an employee either sign or decline the contract or (ii) *repeated contract* which is a long-term contract in which the employer and the employees need to agree on different changes in the future.

In the context of FL, the server is the employer, and the data owner is the employee. The server publishes the contracts and the data owners choose the best-suited contract. The server has limited or no knowledge of resources, data size, and data quality of data owners, which results in information asymmetry. EDs perform the FL tasks and get the rewards in return for services.

C. Game Theory

Game theory is formally defined as a study of strategic interactions among rational individuals [29]. The term 'rational' is referred to as individuals having information about each others' action plan. The game theory aims to help in understanding the situation in which participants interact to determine the probable outcome. A game is any interaction between people in which each persons' payoff is affected by the decision made by others. Game theory has a wide range of applications, including psychology, evolutionary biology, war, politics, economics, and business. Consider two publishers in a city who compete to sell their newspapers. The problem here is to choose a price for the newspaper. Both the publishers are aware of each other decisions. Being rational, they interact with each other to decide the price so that both can benefit.

Game theory consists of four essential elements: players, actions, payoffs, and information [30]. These elements are the rules of the game which are provided by the modeler. The players devise their action plans to maximize their payoff according to the information provided by the modeler. The final set of plans chosen by each player is known as the equilibrium based on which the modeler can predict the outcome of the game. Game theory has two main branches: (i) Cooperative, and (ii) Non-Cooperative. The former deals with the interaction of coalitions or groups when they have the information of payoff only. It aids in the formation of groups and the ways the payment is distributed among players. The players agree to work together to achieve a common goal. Cooperative game theory uses Shapley Value (SV) to determine the contribution of each player to coalition and it divides the payment based on some appealing properties such as fairness, individual rationality, and additivity [31]. SV is calculated by applying several axioms:

- Marginal contribution: The contribution of each player is determined by considering the gain or loss when they are removed from the game.
- Interchangeable players have equal value.
- Non-contributing players have zero value.

- If a game has multiple parts, cost or payoff should be decomposed across those parts.

Non-cooperative game theory deals with the competitive social interactions among players where they aim to achieve their own goals. The players interact with each other to made decisions and when an outcome is reached, that point is known as equilibrium. In game theory, this point is known as Nash equilibrium, which once reached, the players can no longer increase their payoff by changing decisions. At the Nash equilibrium, each players' strategy is optimal, i.e., each player wins because they get the output they desire. To sum up, in the cooperative situation, game theory tells you how to be fair. While, in a non-cooperative situation, game theory tells you how to be smart. In the context of FL, the server acts as the modeler, and data owners act as the players. The server pass on the information to the data owners and they decide their strategic plans to maximize their incentives. The devised strategies are then shared with the server which then decides the data owners to be selected for training and provides them with incentives.

D. Mechanism Design

The mechanism is defined as a specification of message (strategy) space for each individual and an outcome function that maps messages into decisions [32]. Mechanism design is a field of economics and game theory that takes an engineering approach to design economic incentives toward the desired objective in strategic settings. With this approach, we first think of the desired outcome from the system. Based on this outcome, a set of rules are designed that will lead to these desired outcomes. Owing to this design process, mechanism design is also known as *reverse game theory*. The game theory takes the rules of the game and predicts the behavior of participants while mechanism design theory is about making an optimal decision for the rules of the game [33].

Mechanism design deals with the strategic interaction among the participants and the outcome produced by this interaction. One of the major aspects of mechanism design theory is incentive compatibility, this element of providing incentives make the participants act honestly while sharing their private information and act obediently to the plan [34]. Mechanism design involves designing different sorts of assumptions to choose strategies for individuals as a function of their private information. Mechanism design allows economists to analyze and compare the behaviors of markets and institutions that lead to certain outcomes and has broad applications in the management of markets, auctions, and voting procedures. Mechanism design problems have three major characteristics:

- A collective decision problem, for example, privatization of mobile telephony and allocation of spectrum, assigning of work in a group project, and distribution of funding for public universities.
- A standard parameter to determine the participant solutions such as utility, total welfare, and services, etc.
- A description of resources such as individual willingness to pay for the goods.

The two main solution concepts in mechanism design theory are (i) Dominant-strategy equilibrium, and (ii) Bayesian Nash equilibrium. In dominant strategy equilibrium, we are concerned about the optimal choice of participants irrespective of what other participants choose. The dominant strategies provide intense predictions for strategies to be employed by participants [32]. In Bayesian Nash equilibrium, each participants' strategy is the best response to other participants' strategies amid incomplete information. In the Bayesian model, the incentive compatibility constraints are required to behold in expectation only and the participants expect utility maximization only.

E. Auction Theory

Auction theory is concerned with the study and design of rules of interaction for economic transactions to produce some required results using the tools of mechanism design and game theory [35], [36]. In auctions, an auctioneer is a person who wishes to allocate an item for sale to a buyer from a group of bidders. There are four main types of auctions [35]: (i) open cry or English auction, (ii) Dutch auction, (iii) first-price sealed-bid auction (Vickrey auction), and (iv) second-price sealed bid auction. The English and Dutch auctions are based on an iterative process of submitting bids by bidders. In the English auction, bidders keep submitting increasing bids until only one buyer is willing to buy and get the item with the final bid price. Dutch auctions are similar to English auctions except that bidders keep submitting bids with decreasing prices. On the other hand, first-price and second-price sealed-bid auctions are based on one round of bidding. In the first-price auction, bidders submit their sealed bids, and the highest bid wins while in the second-price, the second-highest bid wins.

The most important feature of auction models is the existence of asymmetric information. There are three basic types of auction models: (i) the Private-value model, (ii) the Common-value model, and (iii) the General model. In private value models, each bidder knows the value of an item for sale but that information is private to the bidder itself. In common value models, the actual value is the same for every bidder but bidders have different private information for the actual value. In the general model, each bidder receives a private information signal but each bidders' value is a general function of all the signals [37]. In the context of FL, the server is the auctioneer and the data owners are the bidders. Each data owner submits the bid based on the training cost. Based on these bids, winners are chosen for training and rewarded with incentives.

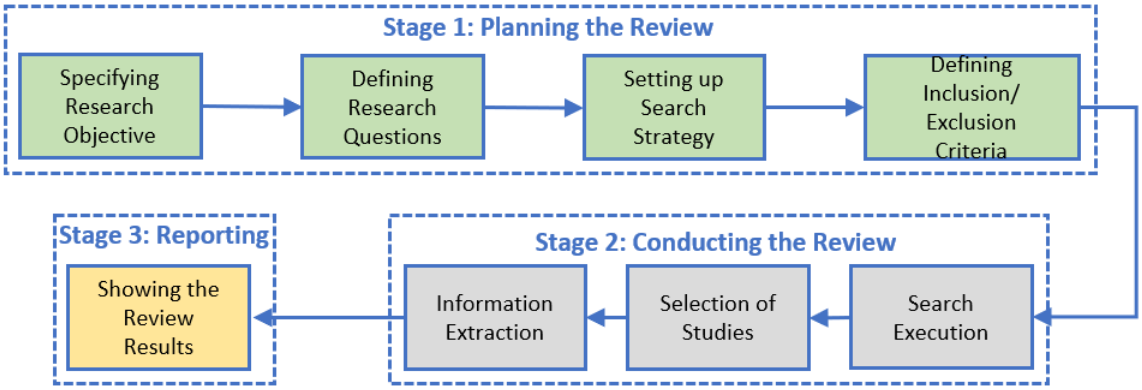
Stage 1: Planning the Review
Specifying Research Objective
Defining Research Questions
Setting up Search Strategy
Defining Inclusion/ Exclusion Criteria
Stage 2: Conducting the Review
Search Execution
Selection of Studies
Information Extraction
Stage 3: Reporting
Showing the Review Results

Fig. 3: The methodology for systematic review.

F. Blockchain

Blockchain was introduced in 2008 [38]. It is a public distributed ledger that keeps a record of committed transactions and this record is stored in a chain of blocks. This technology reduces the risk of fraud and provides a secure and efficient way to transfer assets digitally. It is a decentralized technique and employs cryptographic hash to make it work in a decentralized environment [39]. As blockchain does not require a bank or any other intermediary, it is being used in making Smart Contracts (SCs), online payments, and controlling digital assets [40], [41]. Blockchain provides flexibility to design SCs which play an important role in designing incentive schemes. SCs are just like contracts in the real world but the only difference is they are digital. It is a tiny computer program that is designed to execute contracts in online settings. SCs provides us a way to carry out trusted transaction without needing the third-party. It works in a distributed manner that allows parties to interact and form a contract with each other. One variant of SC is check-point-based SC. The check-point-based SC allows to record and track the behaviors of participants on a database. Using check-point SC, a participant can keep the history of the previous transaction and use it to restore the required information.

In a blockchain, transactions are stored in the form of blocks with each block containing data about the transaction, hash of the current and previous block. These blocks are maintained as a chain of blocks. The transactions are carried out by workers. These workers who create a new and valid block are given incentives for their work, hence the chain is continuously lengthened by them. To create new blocks, Proof of Work (Pow)-based consensus mechanisms are required. There exist many types of consensus mechanisms, such as byzantine fault tolerance based, proof of stake based, and hybrid techniques. In the case of FL, blockchain acts as the server. Both the model and the data owners register themselves with the blockchain and all the transactions are performed by the miners. The blockchain deals with providing incentives to the EDs.

G. Deep Reinforcement Learning

Reinforcement Learning (RL) is the third branch of ML after supervised and unsupervised learning which is targeted at solving sequential decision-making tasks. The major parts of RL are (i) the agent (which takes actions), (ii) the policy (which decides the actions), and (iii) the environment (which is the model of the system involving the state transitions and the reward function). The goal of the agent is to maximize the accumulative reward by deciding a policy for action selection. RL is bound to fail in cases that involve large state and action spaces [42]. As a solution to this, Mnih et al. proposed the combination of RL with DL called Deep Reinforcement Learning (DRL) [43]. They proposed the policy to be represented using Deep Neural Networks (DNNs) hence making the system memory-efficient.

DRL has gained a lot of traction over the past few years. It has now been shown to be helpful in several applications from robotics [44] to networks [45]. Federated DRL was proposed by Zhuo et al. that builds a model of high quality for agents with consideration of their privacies [46]. Recently, DRL has also been shown to solve problems in a federated manner with incomplete information from the different agents. Federated DRL is not the same as multi-agent DRL, as the environment is not the same for all agents which is a common scenario in multi-agent RL. In the case of multi-agent RL, a global state (local states participate to create a global state) is being observed by the agents while in the case of federated DRL the individual users do not share their observations.

III. Systematic Literature Review

This section presents the methodology that we adopted for the systematic review. The motivation for this survey is to systematically review the state-of-the-art mechanisms related to incentivization in FL and associated security challenges. We have conducted a systemic review of a total of 46 research articles that fulfill our inclusion criteria. The review methodology followed in this paper is shown in Figure 3

A. Research Objectives

The following are the objectives of this research:

- **O1:** To present a comprehensive analysis of the existing research on incentivization of FL.
- **O2:** To present a description of security challenges associated with FL especially in the context of incentive schemes.
- **O3:** To identify the pitfalls and limitations of the proposed schemes on incentivization of FL.
- **O4:** To provide insights to the open research issues along with identifying the literature gap in existing research of incentivization in FL.

B. Research Questions

This article will answer the following research questions:

- **RQ1:** Which methods are employed to design incentive mechanisms in FL?
- **RQ2:** What are the security threats to incentivization in FL systems?
- **RQ3:** What are the limitations in the existing incentive mechanisms used in FL?

C. Search Strategy and Searching Phase

1) Databases for searching: To extract the relevant papers, we explored the most common databases and online repositories that include IEEE Xplore, ACM Digital Library, arXiv, Springer, Elsevier, and Scopus. Apart from these, the most common search engine, i.e., Google Scholar was also explored to get the relevant papers. Some of the articles that are used in this paper are published while some of them are pre-print. Most of the content is available online and easily accessible. Considering FL was proposed in 2016, the retrieval time range is from 2016 to March 2021.

2) Search Strings: We used multiple combinations of search strings to get the data from the selected databases. However, the following strings provide the best results

(Incentive schemes for Federated Learning) or (Incentive mechanisms in Federated Learning) or (Incentive mechanism design for FL) or (federated learning nodes selection) or (federated learning challenges) or (federated learning problems) or (Incentivization) or (Incentives) or (security of incentive aware Federated learning) or (security of incentivized federated learning) or (attacks on incentive-driven FL) or (Threats to FL) or (attacks on incentivized FL).

D. Inclusion and Exclusion Protocol

1) Inclusion Criteria: The inclusion criteria were based on the following postulates:

- Article must contain an incentive scheme in the FL environment.
- Article must be related to one of the research questions.
- Article must be related to FL setting.

2) Exclusion Criteria: The following key points were considered for defining the exclusion criteria:

- Articles written in a language other than English.
- Articles that were not available in full text.
- Articles that do not discuss an incentive scheme for FL.

E. Conducting the Review

1) Search Execution: The searching phase was very straightforward. At first, we used the most common search engine, i.e., Google Scholar using the above-mentioned search strings. Secondly, we explored certain databases including IEEE Xplore, ACM digital library, arXiv, Springer, and Elsevier. Finally, we used the above-mentioned strings in the Scopus database query using logical OR and AND operators. The query searches for the specified terms in the title, abstract, or keywords.

2) Selection of Studies: For screening the relevant articles, our screening strategy comprises two steps. In the first step, we reviewed the title and abstract of the extracted papers. With this step, we ensured that there is no repetition of articles as it is very common that an article is published in one of the venues (journals or conference) and is also published on a preprint server, e.g., arXiv to get more visibility, citations, and feedback. Moreover, we also made sure that the selected papers are related to an incentive scheme in the FL settings. However, the title and abstract might not provide a well-defined reflection of the paper. So, after this screening, in the second step, we reviewed the full text of the articles and choose the relevant/non-relevant paper based on our defined inclusion and exclusion criteria. A flowchart representing our searching and screening phase is shown in Figure 4.

3) Information Extraction: After selecting the relevant articles, we read the articles in full detail to determine a basis to categorize these articles. Moreover, the information related to advantages, disadvantages, limitations, goals, approach, system setup, security aspect, and practical results of the work was also considered.

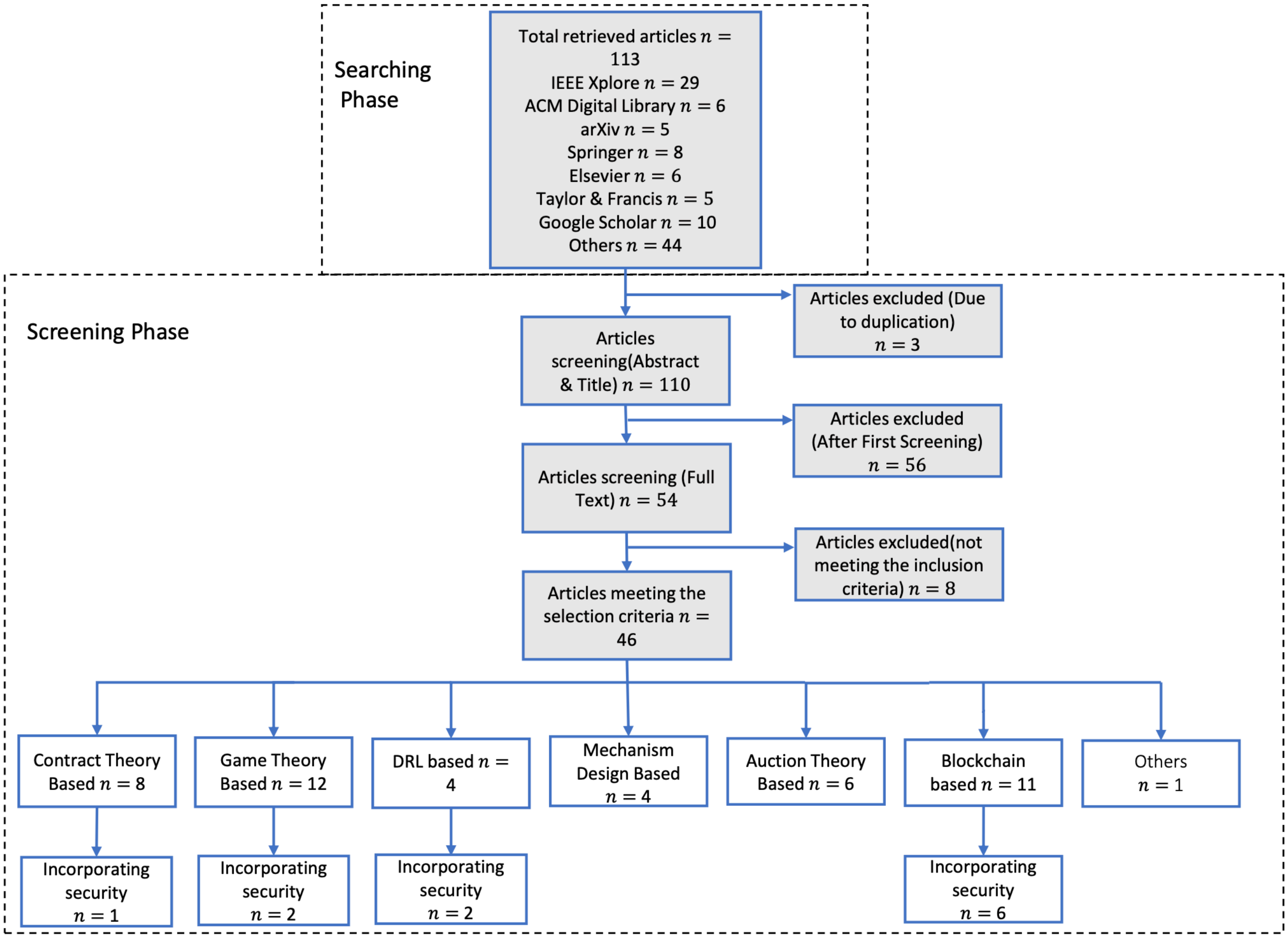
Searching Phase
Total retrieved articles $n = 113$
IEEE Xplore $n = 29$
ACM Digital Library $n = 6$
arXiv $n = 5$
Springer $n = 8$
Elsevier $n = 6$
Taylor & Francis $n = 5$
Google Scholar $n = 10$
Others $n = 44$
Screening Phase
Articles excluded (Due to duplication) $n = 3$
Articles screening(Abstract & Title) $n = 110$
Articles excluded (After First Screening) $n = 56$
Articles screening (Full Text) $n = 54$
Articles excluded(not meeting the inclusion criteria) $n = 8$
Articles meeting the selection criteria $n = 46$
Contract Theory Based $n = 8$
Game Theory Based $n = 12$
DRL based $n = 4$
Mechanism Design Based $n = 4$
Auction Theory Based $n = 6$
Blockchain based $n = 11$
Others $n = 1$
Incorporating security $n = 1$
Incorporating security $n = 2$
Incorporating security $n = 2$
Incorporating security $n = 6$

Fig. 4: Flowchart of screening phase and categorization.

F. Review Results

In this subsection, we will present the statistics related to the outcomes of search results.

1) Overview of the Search and Selection Process Outcome: The search using the aforementioned strategy picked out 113 articles. After title and abstract screening and removing the duplicate articles, the number of relevant articles reduced to 54. These relevant articles are then explored further, we read the full text and it was found that 8 articles do not follow the inclusion criteria. The remaining 46 articles are then categorized further based on their design principle, i.e., contract theory, game theory, deep reinforcement learning, mechanism design, auction theory, blockchain, and others with each having 8, 12, 4, 4, 6, 11, and 1 article, respectively. These articles are then sub-categorized whether they consider security aspects or not. Figure 5 shows the articles divided by their design principle along with the information of associated security.

2) Overview of Selected Studies: The systematic review pointed out 46 articles that are related to incentivization in FL and out of which 11 articles discuss the security aspect as well. We categorized these 46 articles into 7 categories as discussed above. As shown in Figure 6, a majority of these articles are published in journals (45.65%), followed by conferences (23.91%). The percentage of gray literature is (10.87%). Magazine and symposiums have a very small percentage of articles, i.e., (4.35%) and workshops have the lowest percentage at (2.17%). The distribution of selected articles in each venue concerning year is provided in Figure 7, which shows that the topic of incentivization in FL has been very active in the past year. This research area was at its peak in the year 2020 with 35 publications and the majority of these papers were published in the journals. It shows that the researchers are highly interested in the field over the past years. The trend indicates that the majority of articles are published in IEEE, followed by arXiv and Springer. This trend has been shown in Figure 8. Out of the 46 articles on incentivization of FL, only 11 articles explicitly consider the aspect of security, and the majority of it, i.e., (90.90%) have discussed the data poisoning attacks. To defend against attacks, the majority of articles (55.0%) made use of reputation metrics. In addition to the security of incentive-driven FL, a lot of attention has been paid to increasing the communication efficiency, reducing the latency in training and privacy of incentive-driven FL.

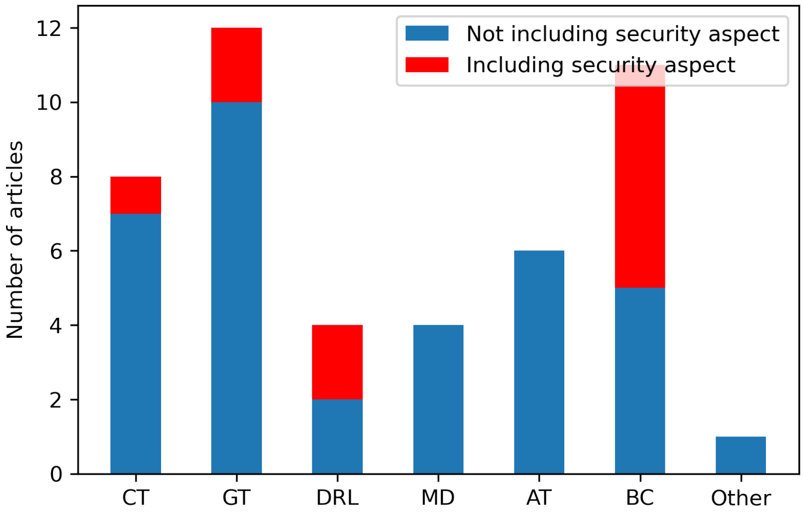

Fig. 5: Distribution of selected publications by design principle. Legend: (CT: Contract Theory, GT: Game Theory, DRL: Deep Reinforcement Learning, MD: Mechanism Design, AT: Auction Theory, BC: Blockchain)

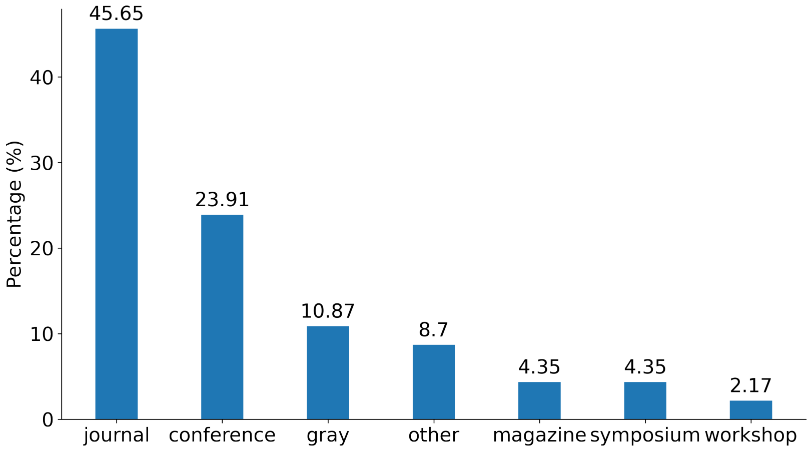

Fig. 6: Distribution of selected publications according to their types.

G. A Discussion on Partially-Related Non-Selected Articles

Ng et al. [53] studied the behavior of FL participants under certain incentive schemes. They propose FedGame, a multiplayer game to analyze how FL participants make decisions under different incentive schemes through crowdsourcing. They evaluate the performance of multiple payoff sharing schemes, i.e., linear, equal, individual, labor union, and Shapley. They do not introduce an incentive mechanism of their own, which is in contradiction to our inclusion criteria.

Ilyas et al. [54] discussed the privacy, management, and integrity of an FL system using homomorphic encryption, blockchain, and integrity mechanism respectively. Due to the non-provision of anything concrete on the proposed scheme, we have not added the article to our review.

Ferrer et al. [55] employed co-utility to tackle the problem of peer honesty. They study decentralized computing in two cases: (i) peers are provided with a task along with the data, and (ii) peer use their data to train a model. They calculate the reputation of their peers but do not discuss how to incentivize them based on this metric. Due to the above-stated reason, this paper does not meet our inclusion criteria.

IV. Incentive Schemes for FL (RQ1)

The incentivization of FL is very important for the survival of the FL community. A lot of attention has been paid to this process over the past few years. It is crucial to provide incentives to the data owners for their services else they will not be

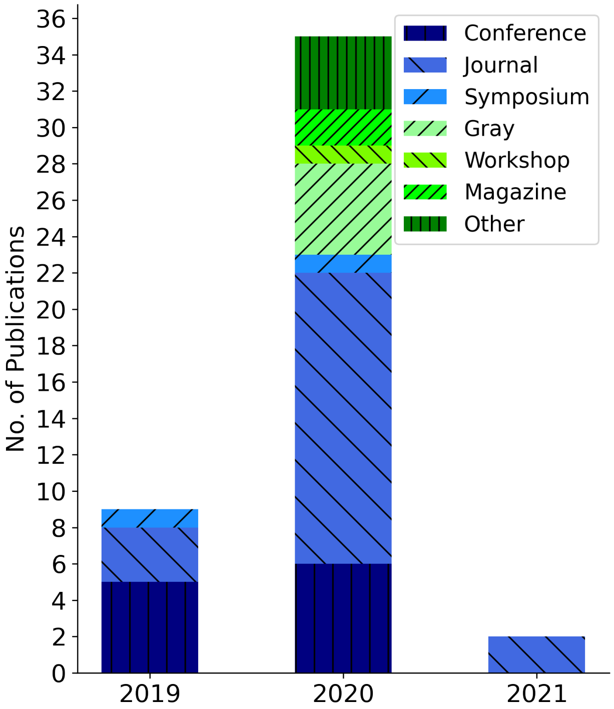

Fig. 7: Distribution of selected publications by types over year.

TABLE IV: Contract theory based mechanisms

Article	Target/Goal	Approach	System setup	Dataset
Kang et al. [20]	Motivating high-quality data owners to participate	Provide incentives based on the data quality and computational resources of the data owners	1 Publisher 100 EDs	MNIST
Ding et al. [47]	Choosing EDs that lie above a certain resource threshold to participate in the task	Incentivizing devices based on the communication delay and computation capacity	1 Publisher 500 EDs	CIFAR-10
Lim et al. [48]	Handling information asymmetry using incentives	Classifying EDs based on the resources and choosing the low-cost EDs to maximize the profit of the server	1 Publisher 5-7 EDs	Task was service providing to the on-ground devices using EDs
Lim et al. [49]	Handling multiple model owners using hierarchical incentive design in IoT settings	Incentivizing data owners based on their resource contributions and incentivizing model owners based on their marginal contributions	Up to 6 Model owners Upto 2000 Workers	Task was of crowd-sensing which had to be done by the workers
Lim et al. [50]	Handling the information asymmetry using the self-revealing property of contracts according to the latency and AoI preferences of the model owner	Incentivizing owners based on their type and resources	50 Workers 1 Publisher	Task was of crowd-sensing which had to be done by the workers
Lim et al. [51]	Handling the information asymmetry using the self-revealing property of contracts according to the latency and AoI preferences of the model owner	EDs are incentivized on the basis of their type and resource usage	Not mentioned	Drones fly over the city to collect data
Lim et al. [52]	Handling multiple model owners using hierarchical incentive design in IoV setting	Incentivizing data owners based on their resource contributions and incentivizing model owners based on their marginal contributions	Upto 6 Model owners Upto 2000 Workers	Task was of crowd-sensing which had to be done by the workers

willing to contribute to the training. Different types of such incentive schemes have been proposed in the literature. In this section, we provide a thematic analysis of thirty-five papers that are focused on designing incentive schemes for FL. These papers are classified into six categories based on a design principle for incentive schemes. These design principles are (i) Contract Theory, (ii) Game Theory, (iii) Deep Reinforcement Learning, (iv) Mechanism Design, (v) Auction Theory, and (vi)

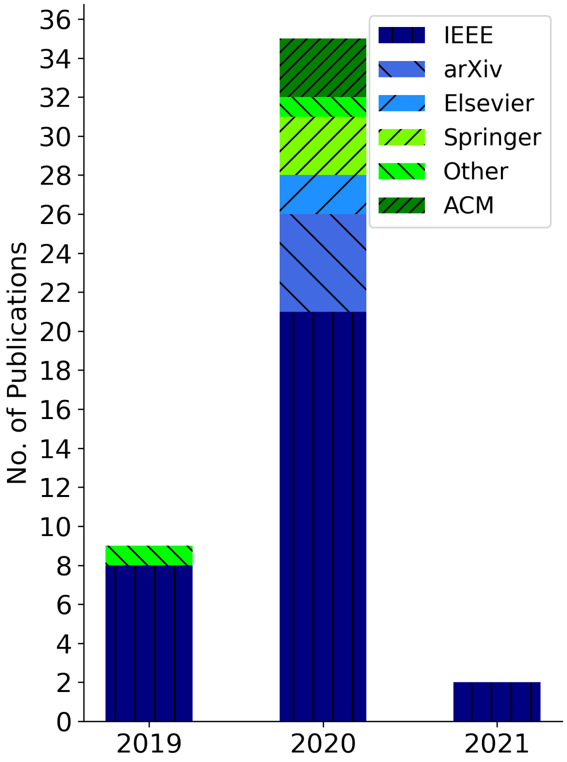

Fig. 8: Distribution of selected publications by publisher over years.

Blockchain. The game theory-based schemes are further divided into Stackelberg games, yardstick competition, and Shapley value-based schemes.

There are two kinds of profits in the discussed papers: an ED's profit and a server's profit. On the one hand, each ED devises the strategy that maximizes its profit (i.e., reward). On the other hand, the server selects the EDs that maximize its profit (i.e., efficient distribution of rewards).

A. Contract Theory Based Mechanisms

This section presents the incentive schemes which are employing contract theory as the design principle. Table IV highlights these incentive schemes providing the details of their goal, approach, and system setup.

To overcome the issue of information asymmetry among the server and the EDs by attracting high-quality data owners, Kang et al. [20] designed an incentive mechanism based on the contributed resources. They define an accuracy-related parameter of local data as the type of the contract mode and each ED selects a contract item to achieve maximum profit. The EDs are first categorized by the quality of their data and then based on their computational resources. High data quality and more computational resources lead to more rewards. The ED selects and signs one of the given contracts and completes the training process. If the ED misbehaves or is not able to finish the training process, it is penalized accordingly. Their results show that the devised incentive mechanism attracts EDs with high-quality data to enhance FL performance and optimize the incentives of both the server and the EDs. By using their proposed mechanism, the task publisher can achieve more profit than the Stackelberg game-based approaches proposed in [56]–[58]. They argue that this lower performance of the Stackelberg game model is due to asymmetric information. Stackelberg game can perform better with symmetric information because the task publisher can optimize its profit owing to knowing the actions of the EDs [57].

Ding et al. [47] argue that the techniques proposed in [14], [59], [60] only consider the server to be able to decide a single dimension which is not always the case and this makes these solutions inflexible. Furthermore, Kang et al. [14] consider the server to have the information about the number of different kinds of users while the other two techniques consider the server to have complete information of the EDs. Ding et al. [47] argue that incentivization becomes difficult when there are networks of heterogeneous devices each with different communication delay and computation capacity, both of which are unknown to the server. As a solution, they proposed an incentive scheme based on two-dimensional information of the EDs, i.e., training cost, and communication delay. The server proposes a contract based on training data size, communication time, and rewards, and sets a maximum communication time for each global round. If the ED wants to participate in the training process, it needs to provide a specific data size for training and must send the updates in less time than the maximum allowed communication time. If the payoff of the ED is negative at the end of the round, it is not allowed to participate in the next round. They study the impact of information asymmetry on the server's profit and show that weakly incomplete information has a little effect on the server's profit while strongly incomplete information is challenging for the server as it is not always optimal for the server to incentivize the group of users with the lowest training cost and delay to participate.

Information asymmetry can cause an incentive mismatch between the server and the data owners. Lim et al. [48] leverage the self-revealing property of multi-dimensional contracts to solve the problem of information asymmetry in Un-manned Aerial Vehicles (UAVs) which are acting as a service. MO is interested in gathering data from a region to predict traffic. The region is further divided into sub-regions to which EDs (UAVs) are assigned. The goal of MO is to maximize its profit by selecting the EDs in such a way that the selected EDs can complete the task at the lowest cost. EDs are classified into different types based on the information provided by them and they are rewarded based on their type. To handle the incentive mismatch between MO and EDs, contract theory is employed which provides guarantees for correct reporting of ED type and allows the server to chose low-cost EDs to maximize its profit.

TABLE V: Game theory based mechanisms

Article	Target/Goal	Approach	System setup	Dataset
Feng et al. [60]	Enhancing the communication efficiency between MO and EDs	The learning services of EDs is priced on the basis of per unit of the training data	1 Publisher 9 EDs	Not mentioned
Sarikaya et al. [59]	Reducing the time delay in each training round	The incentives are provided based on the computing power employed	1 Publisher 20 EDs	MNIST
Pandey et al. [61]	Building a high quality global model along with communication efficiency	Local model accuracy is used a metric to determine the reward	1 Publisher 4 EDs	Not mentioned
Lkhan et al. [15]	Optimization of computational and communication resources	The incentives are provided based on the valuation of updates	1 Publisher 5 EDs	MNIST
Yu et al. [62]	Fair treatment in terms of payoff and waiting time for payment	Incentivizing EDs based on their contribution to the model	1 Publisher 100 EDs	Not mentioned
Hu et al. [63]	Motivate data owners to actively participate in the training with privacy guarantees	MO provides incentives based on its privacy budget	1 Publisher 100 - 1000 EDs	Not mentioned
Qu et al. [64]	Privacy protection of trained models	Rewards are calculated based on the contribution of EDs	Not mentioned	CIFAR-10
Lee et al. [65]	Reducing the latency in the FL process	Based on the services provided by MECs, incentives are calculated	1 Publisher Multiple MECs	Not mentioned
Sarikaya et al. [66]	Reducing the training time for each batch in a synchronous setting	Reward is determined on the basis of deviation of each ED from target delay time	1 Publisher 10 EDs	MNIST
Song et al. [67]	Reducing the computational cost of the server	Reward is provided based on contribution index	1 Publisher 5 EDs	MNIST

In crowd-sensing, the presence of insufficient data samples makes it difficult to train efficient ML models. Lim et al. [49] argue that the previously proposed schemes only consider a single MO which may not be always the case. They proposed a hierarchical incentive scheme for FL using contract and game theory to facilitate collaborative ML among multiple MOs. Their proposed scheme deals with two types of incentive discrepancies: (i) data owners with MOs, and (ii) MOs with each other. In the proposed system, data owners gather data and upload it to the MO where the ML model is trained. These MOs then collaborate and upload local model parameters to a third-party server where the aggregation is performed. For the case of incentive discrepancies of data owners with MOs, they employed a contract theoretic approach to maximize the reward of the data owner based on their type. In the case of incentive discrepancies of MOs with each other, the authors utilize game theory to reward the MOs based on their marginal contributions. This scheme is hierarchical as in the upper level the payoff received by MOs using game theory directly affects the reward of data owners in the lower level contract theoretic approach. A similar approach has been discussed by Lim et al. [52], the authors proposed the usage of collaborative FL among multiple MOs on the Internet of Vehicles (IoV) setting.

Lim et al. [50] designed an incentive scheme exploiting contact theory that can be adjusted according to the latency and Age of Information (AoI) preference of the MO. They make use of the self-revealing property of contract theory and design a mechanism that rewards EDs based on their updated cost and type, amid information asymmetry. They compare the proposed scheme with two different pricing schemes: (i) *uniform pricing scheme* in which equal reward for all types of EDs in the information asymmetry setting, and (ii) *discriminatory pricing scheme* in which all the EDs' types are known, i.e., information asymmetry does not exist. The results show that MO achieves higher profit in the proposed scheme in comparison to the uniform pricing scheme but less profit than the discriminatory pricing scheme. A similar approach has been discussed by Lim et al. [51] for FL deployed on a server with EDs being UAVs acting as service providers.

Synthesis and reflection: Contract theory-based mechanisms are best suited for the scenarios when complete information asymmetry exists in the system. These schemes usually take into consideration the computational and communication resources of EDs and then decide the rewards. This is an effective way to design incentive schemes because a majority of the cost incurred by EDs is due to the training of data and transmission of models. Contact theory-based mechanisms are better as compared to game theory-based mechanisms amid information asymmetry.

B. Game Theory Based Mechanisms

This section discusses the incentive schemes based on game theory as the design principle. Table V highlights these incentive schemes providing the details of their goal, approach, and system setup.

1) Stackelberg Game-based Schemes: Feng et al. [60] argue that the direct communication between the MO and EDs to exchange model parameters is inefficient. They proposed to adopt a relay network for the construction of a communication platform. They formulate a Stackelberg game to study the interaction between the EDs and the MO. They consider a system consisting of two parties: (i) the MO which publishes the FL task and handles the aggregation of local updates, and (ii) the EDs working as learning service providers. Along with the transmission power, the EDs also have to choose the relay node due to coupled interference of wireless networks. In the game, the MO acts as a buyer while mobile devices act as sellers. Mobile devices decide the price for learning service in the form of one unit price for the training in the non-cooperative game. In return, the MOs decide the amount of training data for each mobile device. Under such circumstances, the mobile devices seek to maximize the profit by optimizing the price of learning data such that MO chooses a larger size of training data.

Sarikaya et al. [59] argue that it is not fair to deal with all the EDs identically as few of the EDs can also be selfish. Furthermore, they discuss another problem that is faced in the case of synchronous batch implementation tasks where we have to deal with strangler effect, i.e., the high-performing EDs have to wait for the low-performing EDs to submit their results to get their incentive. Sarikaya et al. [59] developed an incentive scheme focusing on reducing time delay in each training round. In the proposed setting, the relation between MO and participants is modeled using the Stackelberg game where MO is the buyer and participants are sellers. The MO negotiates with the participants about computing power and based on this, provides them with incentives.

Pandey et al. [61] designed a crowdsourcing framework that focuses on increasing communication efficiency which is a key challenge in designing an FL framework. The interaction between Mobile Edge Computing (MEC) server and EDs is modeled using a two-stage Stackelberg game. The reward is provided based on the accuracy of the local models, i.e., $/accuracy level. In the first stage, the server offers a uniform reward rate. In the second stage, the EDs devise a strategy to improve the local accuracy based on provided reward rate. The authors claim that the proposed scheme outperforms the baseline approach, which considers the worst response by the EDs to reach a consensus accuracy with offered reward rate, by 22%, however, they limited their model by considering uniform pricing strategy, i.e., all the EDs will get the same reward for a given accuracy. This scheme is significant in terms of fairness but the server's utility could be further improved by employing a differential pricing strategy.

Optimization of both the computation and communication resources is required for an FL pipeline to work properly. As an extension to their previous work [68], Lkhan et al. [15] proposed a Stackelberg game-based incentive scheme to effectively model the interaction between EDs and the server in edge networks. The server acts as a leader and offers a reward rate while the EDs act as followers and submit their best updates. The server evaluates these updates, offers an incentive, aggregates the global model, and broadcasts the reward rates while optimizing its utility. Strategies are decided by the EDs based on these rewards. They show that the increase of the reward rate motivates the EDs to perform more iterations in one global round hence increasing both the local and global accuracy.

Hu et al. [63] argue that the incentive schemes proposed in [20], [61] are not practical as all the burden of maintaining the privacy of the local models is put on the server. Such practices might reduce the willingness of the high-quality EDs to participate in the training process. They propose a two-stage Stackelberg game approach to design an incentive mechanism. Their proposed mechanism motivates data owners to actively participate in the FL training process with meticulous privacy guarantees by defining a specialized privacy budget. The goal of the EDs is to maximize their utility by selecting an optimal privacy budget and the goal of the MO is to maximize its utility by selecting the rewards. In the first stage of the game, the server declares the total reward while in the second stage, the users strategize their privacy budget in a non-cooperative game to maximize their utility. After the successful calculation of the payment, the users with positive payments train the models, add a random noise (dependent on the privacy budget) to the models, and send these local updates to the server. The addition of random noise preserves the privacy of the local models and makes it almost impossible to derive information about the private features of the data.

Previous works on market analysis were confined to only one MEC operator. With a focus on latency reduction, Lee et al. [65] proposed a distributed market model consisting of IoT devices, multiple MEC operators, and a cloud operator. They modeled the interaction among MECs and cloud using as Stackelberg game in which the MEC operators maximize their payoff by incorporating a tradeoff between revenue and energy consumption. In the proposed framework, the MO coordinates with IoT sensors to efficiently distribute the sensing data over multiple MECs. These MECs receive a shared model from the cloud, locally train the model, and upload these local models to the server hence creating a global model. MECs acting as leaders decide the service prices while MO acting as followers choose the best response strategy for the prices. They argue that the proposed method is guaranteed to reach a Stackelberg equilibrium solution on a well-defined utility function that maximizes the payoff of all participants. However, they assume that the cloud operator is in close vicinity of the IoT devices and it has a direct wireless connection to MECs which is an unrealistic assumption.

TABLE VI: Deep Reinforcement learning based mechanisms

Article	Target/Goal	Approach	System setup	Dataset
Zhan et al. [69]	Tackle the AoI by evaluating the contribution of EDs	Incentives are provided based on the contribution of EDs	1 Publisher 10 EDs	Not mentioned
Zhan et al. [70]	Reducing the time for training	A DRL based mechanism uses history of EDs to determine the incentives	1 Publisher 5 Edge nodes	UMass Transit data from buses

2) Yardstick Competition based Schemes: Sarikaya et al. [66] highlighted that despite the benefits of their previously proposed Stackelberg game-based approach [59], it increases the training time for each round. It provides higher incentives to the lagging EDs and hence, EDs prefer to utilize fewer resources which is not acceptable. As a solution to this, in [66] they proposed a scheme based on yardstick competition. The proposed scheme is an incentive mechanism that focuses on reducing the training time for each batch in a fully synchronous stochastic gradient descent setting. A yardstick is used to calculate the desired delay in each round and reward is determined based on the deviation of each ED from this yardstick. The MO announces a yardstick, i.e., the desired target delay in which each ED should complete the local training. If the ED completes the training at the time exactly equal to the yardstick, it receives a fixed reward. The reward increases proportionally as the training time reduces and decreases as the training time is greater than the yardstick. They show that the proposed scheme reduces the delay with the increase in the reward rate. However, they consider the delay to be only dependent on the CPU power allocated by ED's but in practical cases, several factors impact the delays which make this system inflexible.

3) Shapley-Value based Schemes: Qu et al. [64] designed a two-phase ML framework for the incentivization of edge servers in a cloud-edge-device cooperative manner. The first phase deals with the FL process between cloud and edge servers while the second phase deals with global model segmentation for personalized requirements. In the FL process, the interaction between cloud and edge servers is modeled as a cooperative game and SV is used to fairly distribute the rewards [71]. Rewards are allocated based on the contribution of edge servers. To evaluate the contribution of edge servers, gradient values are compared. If the gradient value obtained by the edge server is closer to the global update value, it indicates that the edge server has contributed more to finding the best parameters. Although the proposed framework provides a fair distribution of incentives to the edge server it does not consider the incentivization of the connected ED's for providing the data.

The allocation of profit to EDs has been a major problem in FL. An important aspect of this problem is to determine a metric to quantify the contribution of EDs. SV-based schemes, as proposed by [72], are common in game theory to evaluate the contribution of EDs, however, it imparts extra cost to the server. To tackle this problem, Song et al. [67] devised a new metric based on SV named Contribution Index (CI) for the valuation of contribution. They proposed two gradient-based methods to determine the CI. The reward is distributed based on the CI. They showed that the proposed method is remarkable, however, it is limited and can only work for Horizontal FL.

4) Others: As the training of the ML models takes time, the EDs have to wait before they are paid. Yu et al. [62] proposed an effective payment scheme called federated learning incentivizer (FLI) by keeping in consideration the importance of time delay. FLI is a real-time algorithm that computes payoff in installments for data owners. The goal of the system is to maximize the utility by dividing the given budget among EDs while treating them fairly in terms of payoff and waiting time. The server keeps track of the amount needed to be paid and is referred to as regret. They compare their proposed scheme with five state-of-the-art schemes: namely, linear, equal, individual, union, and Shapley's scheme based on SV as proposed in [31]. They show their scheme achieves the highest possible revenue.

Synthesis and reflection: Game theory-based schemes are focused on optimizing the resources (latency, computational, and communication resources) while providing the incentives. In these mechanisms, the server negotiates with the EDs about their resources and the rewards are decided accordingly. The Stackelberg game-based schemes provide an adequate optimization of resources but on the other hand, it increases the training time for each round. In addition to Stackelberg's game-based mechanisms, there exists promising work that made use of SV to determine the contribution of EDs and fairly allocate the rewards to them.

C. Deep Reinforcement Learning Based Mechanisms

This section discusses the incentive schemes which are using DRL as the design principle. Table VI highlights these incentive schemes providing the details of their goal, approach, and system setup. Now, we will discuss all of these schemes in detail.

Due to the incomplete information about the decisions of the other EDs, the EDs are not able to come up with an optimal decision. Also, it is difficult to evaluate the contribution of an ED to the accuracy of the global model. Due to these concerns, it is difficult to devise an optimal incentive scheme for the EDs which leads to low participation by the EDs in the training process. Zhan et al. [69] designed a DRL-based incentive mechanism to devise an optimal pricing strategy for the server along with the optimal training strategy for the EDs. The main objective of this mechanism is to tackle the AoI by evaluating the contribution of participants. First, they formalize the problem as a Stackelberg game and show the uniqueness of the Nash equilibrium that describes the steady-state of the whole FL system, given the full knowledge of participants' contributions. Then they propose a DRL-based scheme to tackle the issue of unshared information. DRL can learn from states using historical training records. The target of the server is to minimize the total reward while each ED's goal is to maximize its revenue, i.e., the difference of received reward and the cost of training and data collection. The server observes the past payment strategy along with past participation history and devises a payment strategy. The EDs interact with each other to determine their participation strategy under this payment strategy. Each ED updates the local models and receives a reward based on training and payment strategy. Furthermore, a comparison was made with random and greedy approaches which showed that DRL-based servers can achieve a better utility than the other two. The mechanism of the DRL in their scheme has been shown in Figure 9.

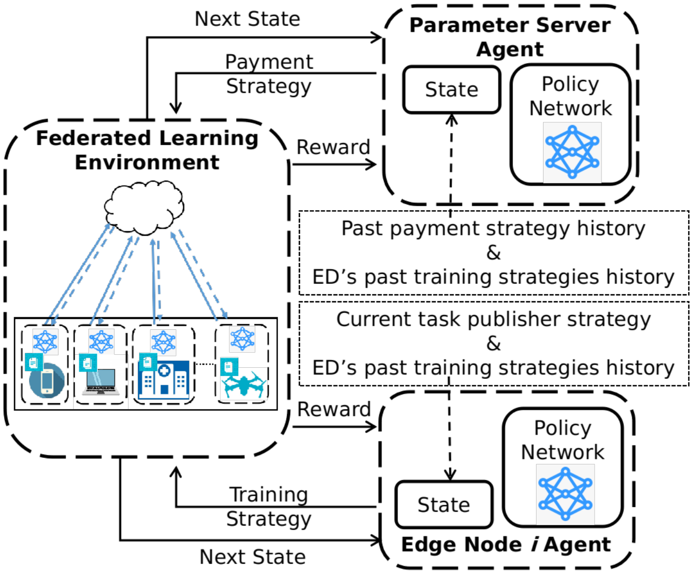
Next State
Parameter Server Agent
Payment Strategy
State
Policy Network
Federated Learning Environment
Reward
Past payment strategy history & ED's past training strategies history
Current task publisher strategy & ED's past training strategies history
Reward
Policy Network
State
Training Strategy
Next State
Edge Node *i* Agent

Fig. 9: The working of DRL in the game-theory based incentive-aware FL scheme proposed by [69] as shown in their paper

Deciding an optimal strategy for incentivization in a dynamic environment with limited or no knowledge of the EDs is a challenging problem to be solved using mathematical analysis. To this end, Zhan et al. [70] proposed an incentive mechanism to maintain the trade-off between reducing the learning time and maximizing the payment in FL using DRL. The incentive of each of the EDs is decided in a hierarchical game environment whose optimal solution is found using DRL. The server acts as the agent in the proposed DRL setup and learns the pricing strategy based on experience rather than having information about the data owners. The server acts as the main decision-maker for the price while EDs define the limitations, rules, and rewards.

Synthesis and reflection: DRL-based schemes benefit both the MO and EDs. It allows a better interaction among EDs to devise optimal strategies to decide their resources while allowing MO to design an efficient pricing strategy. The key focus of these mechanisms is usually to reduce the training time. DRL based schemes work best amid information asymmetry because it decides the strategies based on history (i.e., experience).

D. Mechanism Design Based Mechanisms

This section presents the incentive schemes which are employing mechanism design as the design principle. Table VII highlights these incentive schemes providing the details of their goal, approach, and system setup.

TABLE VII: Mechanism design based mechanisms

Article	Target/Goal	Approach	System setup	Dataset
Cong et al. [73]	Fair sharing the incentives among the EDs	The incentives are provided based on cost type and data quality of the EDs	Experimental evaluation is carried out using hypothetical specifications	
Kim et al. [74]	Providing guarantees for users' privacy	A valuation function based on EDs contribution determine the incentives‘	10 Publishers 50 EDs	Not mentioned
Cong et al. [75]	Encourage EDs to truthfully report their cost type	Incentives are provided on the basis of reported cost type and data quality	Performed the evaluations for hypothetical scenario	

TABLE VIII: Auction theory based mechanisms

Article	Target/Goal	Approach	System setup	Dataset
Le et al. [77]	Motivating EDs to contribute for the FL training	Reward is provided on the basis of cost of energy	1 Publisher 10 - 50 EDs	Not mentioned
Zeng et al. [78]	Reduction of computational overhead along with attracting high quality data owners	Incentives are distributed based on resources employed for training	1 Publisher 100 EDs	MNIST CIFAR-10 HPNews
Jiao et al. [79]	To reduce the communication traffic congestion and commercializing FL services	EDs submits the information of data quality, communication and computational capacity and rewards are determined on the basis of these parameters	1 Publisher 50 EDs	MNIST
Le at al. [80]	Encouraging EDs to participate in FL training process	Rewards are provided based on energy cost while trying to minimize social cost	1 Publisher varying number of EDs	Not mentioned
Ying et al. [81]	Attracting the EDs to participate while preserving the privacy	A distribution probability is used to select EDs and reward provided based on execution price	1 Publisher 80 EDs	KDD-Cup-99
Roy et al. [82]	To reduce the computational latency and network congestions	EDs are incentivized based on the quality of completed task	10-50 Publishers 10-50 EDs	Not mentioned

Cong et al. [73] proposed a scheme "FVCG" based on Vickery-Clarke-Grooves (VCG) mechanism [76] for optimally and fairly sharing the incentives among the EDs. Their goal was to maximize the social surplus while minimizing the unfairness of the federation. They defined 5 objectives of FVCG: (i) *Dominant Incentive Compatibility (DIC)*: All EDs truthfully report their data qualities and cost types; (ii) *Individual Rationality (IR)*: No ED gets worse off than if it quits the federation; (iii) *Weak Budget Balance (WBB)*: For all feasible data quality vectors and cost type vectors, the sum of payments to all data owners is no greater than the federation revenue; (iv) *Social Surplus Maximization (SSM)*: The social surplus is maximized; and (v) *Fairness*: Fairness is attained by minimizing an ED-defined unfairness function. Their experimental evaluation is carried using hypothetical specifications and as stated in the article, cannot be applied to real scenarios. They argue that their results are enough for revealing the reasonableness of FVCG and their neural network-based method.

Similarly, Kim et al. [74] also proposed the use of the VCG mechanism for an incentive scheme with the focus on guaranteeing users' privacy. The proposed system consists of a set of MOs and a set of EDs in which each MO is associated with some IoT devices and has its budget. The interaction among the operator and IoT devices is modeled as a game where the valuation function of each ED represents its payoff based on its contribution. Laplacian noise is added to the model before being uploaded to the MO to keep the models uploaded by the EDs private. A similar target is achieved in [15] where the major difference lies in the design of the incentive model. [15]'s scheme is based on the non-cooperative game model, and their incentive model is designed based on the VCG mechanism, which is the opposite concept of the non-cooperative game approach. Their scheme outperforms the schemes proposed in [49], [62], [67].

Cong et al. [75] employed mechanism design to tackle the problem of incentive mechanism in FL. They divide this problem into two sub-problems: demand-side problem and supply-side problem. To solve the former, they introduced the Cremer-Mclean mechanism, and the latter is solved by developing the VCG based mechanism, Procurement-VCG (PVCG). The proposed mechanisms encourage the EDs to submit their cost type truthfully and also provide the best quality data for training. These mechanisms also provide theoretical guarantees for incentive compatibility, individual rationality, and budget balance.

Synthesis and reflection: Mechanism design-based schemes mainly uses VCG mechanism for designing the incentives schemes. VCG is an effective mechanism that takes into account certain critical objectives. Therefore, the incentive schemes based on VCG outperform several other schemes.

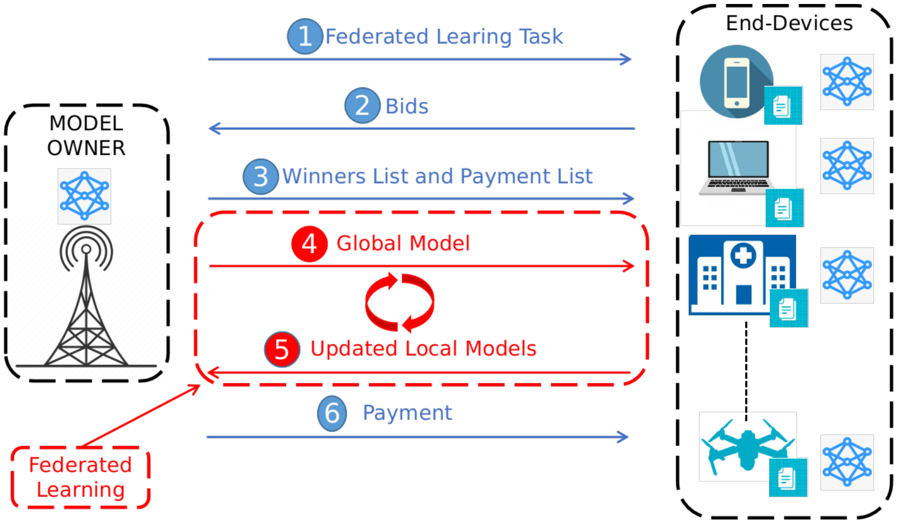

Fig. 10: An overview of the auction-theory based scheme proposed by [77] as shown in their paper

E. Auction Theory-Based Mechanisms

This section discusses the incentive schemes based on auction theory as the design principle. Table VIII highlights these incentive schemes providing the details of their goal, approach, and system setup. The following discussion will elaborate on these schemes.

Le et al. [77] proposed the usage of auction theory as an incentive mechanism using primal-dual greedy auction mechanism for deciding the winner of the auction. The server acts as an auctioneer and EDs act as the sellers. In this setting, each ED submits its bid based on the energy cost and the goal is to choose a winner and maximize social welfare, i.e., the system's utility. The server first publishes the auction rules and the FL task and each ED calculates the number of resources required. Based on these resources, EDs submit bids that include the required amount of resources, local accuracy, and energy cost of training. The winner is selected in the social welfare maximization problem which is an NP-hard problem [83]. It is solved using a primal-dual greedy approximation algorithm. The algorithm updates both primal and dual variables in iterations. The server categorizes the bids in decreasing order of the normalized values of bids. This is followed by determining the critical value, i.e., the minimum value end-user has to bid to win the requested resources. Based on this value, the winner is selected and it participates in the FL process, and receives payment. Their experiments show that their proposed scheme can generate a 400% larger social welfare than the fixed price scheme. An overview of their scheme is shown in Figure 10.

Zeng et al. [78] proposed an incentive scheme "FMore" with multi-dimensional procurement auction for a multi-dimensional MEC setup. Their proposed approach has a low computational overhead and encourages high-quality data owners to participate in the training process at a low cost hence improving the training quality. They tested their proposed scheme using simulations and real networks and show that this scheme can achieve a drastic performance improvement in comparatively fewer training rounds.

Jiao et al. [79] presented the first auction mechanism for the FL community which applies the Graph Neural Networks (GNN) and DRL in the design of a truthful auction mechanism to solve a combinatorial NP-hard problem. They formulate two auction mechanisms: (i) an approximate strategy mechanism which guarantees the truthfulness (EDs are honest while declaring their data types), individual rationality (EDs incur no loss during incentivization), and computational efficiency; and (ii) an automated scheme based on DRL and GNN to further improve the social welfare (utility). They developed a market model based on auction theory to commercialize the FL services among different organizations. The overall goal of this system is to maximize social welfare. The server employs a Reverse Multi-dimensional Auction (RMA) mechanism to chose the optimal set of EDs and decides payment. In the RMA mechanism, the data owners are first divided into G groups based on the quality of data they own. Then it selects K-data owners using randomized and greedy ways from the groups and calculates the payment for each worker. RMA provides the guarantees for truthfulness, individual rationality, and computational efficiency but the maximization of social welfare is still restricted due to randomization and channel conflicts among data owners. The authors used GNN to construct Spectrum Conflict Graph [84] to tackle channel conflicts and produced effective embeddings. They adopted the deep Q-learning [85] that solves the maximization of social welfare. An overview of the DRL part of the proposed scheme has been shown in Figure 11.

Le et al. [80] proposed an auction-based mechanism that can provide guarantees for three parameters: truthfulness, individual rationality, and efficiency. The server acts as a seller in the auction framework while the data owners act as the buyers. Each data owner determines the energy cost to be incurred for training and then submits a bid to the server. The submitted bid includes the CPU cycle frequency, the communication power, and the cost. Based on these bids, the server chooses a different

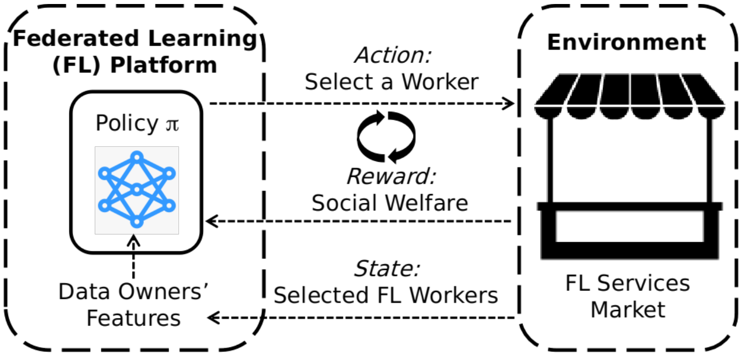

Fig. 11: An overview workflow of the DRL part of the auction-theory based FL scheme proposed by [79] as shown in their paper

set of winners for different global rounds and calculates the rewards to minimize the social cost. The problem to minimize the social cost is modeled as an NP-hard problem and a randomized auction mechanism is used to solve it.

Ying et al. [81] developed an auction theory-based incentive scheme in the mobile crowdsourcing setting. The proposed system is termed SHIELD and is focused on truthfulness and individual rationality. It preserves the privacy of the model updates and bids of EDs using differential privacy. The designed system consists of two stages: (i) Incentive Procedure, and (ii) Training Procedure. In the first stage, the MO publishes the training task for ED's. After analyzing the given task, each ED submits its bid which is the price of executing the task. According to received bids, the MO designs a probability distribution to select a set of winners. As part of the second stage, the winning EDs train the model and receive the payments. By using their proposed system, the MO bears less cost and payment as compared to the baselines. They considered the cost of training to be proportional to data quantity which is somehow a valid assumption because the majority of the cost is due to the consumed resources for data training, however, neglecting the communication cost makes this system a bit inflexible.

To reduce the computation latency and network congestions, Roy et al. [82] proposed a Mobile Device Cloud (MDC) environment exploiting auction theory. They designed a reputation-based auction model termed FEST containing buyer and worker devices. Buyer devices collaborate to determine the reputation of workers using FL. The worker devices are selected using the reputation metric and are incentivized based on the quality of the completed task.

Synthesis and reflection: Auction theory-based schemes focus on attracting high-quality data owners and reducing the latency, computational and communication resources. Schemes based on the design principle can provide guarantees for some crucial parameters such as truthfulness, individual rationality, and computational efficiency and that's why they perform well in incentive-driven systems. In these schemes, the problem to minimize the social cost is usually modeled as an NP-hard problem and several mechanisms are employed to solve this problem.

F. Blockchain Based Mechanisms

This section discusses the incentive schemes which are using blockchain. Table IX highlights these incentive schemes providing the details of their goal, approach, and system setup. Now, we will discuss all of these schemes in detail.

To tackle the issue of deciding the reward for the honest EDs in an FL-based system, Toyoda et al. [86] proposed a strategic mechanism design using blockchain. In the proposed setting, the task publisher publishes the FL task along with a model description, starting time, number of data owners in each round, reward in each round, and overall total reward. In each round random EDs are chosen and to access the models of the previous update, a key is shared with the selected data owners. The reward is distributed based on the votes cast by each ED for the top models in the previous round. In the last round, the reward is equally distributed among the data owners since there are no data owners to vote. But this equal division is itself a problem since data owners can submit the same update again and get the reward by fraud. To handle this, the task publisher keeps the information of a total number of rounds private and the data owners have no idea that this is the last round of training.

Zhang et al. [87] proposed a smart contract-based mechanism for device failure detection which is an essential problem in Industrial IoT (IIoT). In the proposed mechanism, each device builds a Merkle tree where each leaf node represents a record of data collected by the ED. The root of the Merkle tree is stored on a blockchain, which ensures the integrity of data, at a pre-conFigure d interval. To motivate high-quality data owners to participate in the training process, they reward the EDs with tokens based on the size and centroid distance between two classes of the data used in training. For each round, the central server chooses a fixed number of EDs which train the models and upload them to the central server. Through experimentation in a real industry failure detection use case, they showed that their proposed mechanism motivates EDs to provide sufficient resources for model training while reducing the impact of data heterogeneity, and preserving EDs' data privacy and integrity.

Liu et al. [72] proposed a blockchain-based Peer-to-Peer (P2P) payment scheme called *FedCoin*. The incentives are provided using a blockchain network that works in connection to the FL network. After the uploading of the local models by the EDs,

TABLE IX: Blockchain based mechanisms

Article	Target/Goal	Approach	System setup	Dataset
Toyoda et al. [86]	Design a mechanism to force EDs to act rationally	The reward is distributed based on the votes casted by each ED for the top models in the previous round	Performed the theoretical analysis only	
Zhang et al. [87]	Reducing the impact of data heterogeneity and preserving EDs' data privacy and integrity	Reward the EDs based on the size and centroid distance between two classes of the data used in training	1 Publisher 4 EDs	Performed the experiments on hotel AC systems by collecting their own data
Liu et al. [72]	Encouraging high quality EDs along with reducing computational resources	Rewards are provided based on training price	1 Publisher 100 EDs	MNIST
Hieu et al. [88]	To achieve a high accuracy along with minimizing the energy consumption and training time	Incentives are provided based on data and energy consumption	Not mentioned	Not mentioned
Kumar et al. [89]	Enhancing privacy	The validation score forms the basis to develop a criterion to provide incentives	1 Publisher 5 EDs	MNIST
Rehman et al. [90]	To reduce the problem of coarse-grained predictions	Reputation metric is employed to distribute incentives among EDs	Not mentioned	Not mentioned

the FL server updates the global model and publishes a task of SV calculation to the blockchain server with the processing fee. The total budget for each round of training is divided into three parts: (i) TrainPrice, (ii) SapPrice, and (iii) ComPrice. The consensus nodes in the blockchain then determine the SV's and choose a winner block. This block receives payment of summation of train price and sap price from the server. The training price is divided among the EDs based on their SV, by creating transactions in the blockchain. The sap price is paid to the blockchain miner for the calculation of SV of the EDs.

Hieu et al. [88] proposed a DRL-based incentive scheme for blockchain-based FL whose goal is to achieve a certain accuracy along with minimizing the energy consumption and training time. They consider an FL network consisting of a MO and mobile devices including data owners and blockchain miners. The MO first publishes a request to devices containing an initial global model and a record of payments for participants and miners. The payment record indicates that how much data and energy should be used by participants. Moreover, it also reveals the rate of block generation used by miners. The participants train the model locally, generate a transaction, and convey it to the miner. All the received transactions are stored in the miner's queue for cross verification. The miners solve a cryptographic problem and the first miner to solve it is the "winning miner" which is chosen as authority to create a new block. This winning miner adds some transactions into the new block and sends them to other miners too. The miners verify the transactions and add them to their local blockchain. Finally, the MO aggregates the transactions and creates a new global model. This process is repeated until the desired accuracy is achieved.

The security of transactions taking place in the FL framework is of great importance. Considering the secure nature of the blockchain technology, Kumar et al. [89] employed Ethereum (blockchain technology) along with InterPlanetary File System (IPFS) to create a decentralized environment for FL. Ethereum smart contracts [91] are used to create a value-driven incentive mechanism. In the proposed setting, MO publishes the FL task along with a small dataset for the evaluation of the model's performance. The global model is distributed to EDs using smart contracts. The updates provided by the EDs are then validated by miners using the validation dataset and each update is assigned a validation score. The validation score forms the basis to develop a criterion to provide incentives. If the validation score is higher than the validation threshold, the update is considered to be included in the aggregation process. The EDs are rewarded with ether, i.e., the cryptocurrency of the Ethereum platform, based on the validation scores.

Rehman et al. [90] proposed a fine-grained FL framework to overcome the problem of coarse-grained predictions which are not required in applications involving personalized prediction services. To ensure trustworthy collaborative training in MEC systems, they propose a blockchain-based reputation-aware fine-grained FL framework. A reputation-based incentive scheme is proposed to encourage EDs to actively participate and provide high-quality data. The proposed scheme maximizes the benefits of honest EDs while minimizes the benefits of dishonest EDs.

Synthesis and reflection: Blockchain-based incentive schemes usually focus on enhancing the integrity and privacy of FL systems. Blockchain provides a secure means of committing transactions among the MO and EDs. Also, it enables smart contracts-based systems which are very effective for designing and distributing rewards.

G. Others

To address the problems of aggregation of local models and reducing communication overhead cost, Wu et al. [92] proposed a Federated Mediation (FedMed) framework which employs an adaptive aggregation scheme along with a mediation incentive

TABLE X: Pros and cons of incentive schemes based on design principle.

Design principle	Advantages	Disadvantages
Contract theory-based schemes	• Tackle the issue of information asymmetry • Can handle multi-dimensional information to design incentive schemes • Motivates EDs to report their correct types as per users' preference	• Focused on optimizing the rewards of EDs only
Game theory based schemes	• Provides better optimization of resources • Fair distribution of rewards	• Poor performance amid information asymmetry • It cannot incorporate multiple factors to design incentive schemes
Deep Reinforcement Learning based schemes	• Tackle the issue of information asymmetry • Motivates high quality EDs to participate at low cost	• High computational cost
Mechanism design based schemes	• Optimal and fair distribution of rewards • Motivates EDs report their data and cost type truthfully	• Perform well only when the MO has full knowledge of EDs (information symmetry
Auction theory based schemes	• Provide guarantees for truthfulness, individual rationality and efficiency • Reduces the computation latency	• Biased to maximize the utility of MO only
Blockchain based schemes	• Preserves EDs privacy and integrity • Provides a secure mean to commit transactions • Provides an efficient way to keep track of transactions	• Poor interaction of MO and EDs to determine the rewards • Introduces the time delays in the system

scheme. The mediation incentive scheme is used to select the aggregation algorithm, i.e., either adaptive aggregation or FedAvg aggregation. They proposed an incentive scheme based on the difference of the loss of the current and previous round. They argued that the model trained in the proposed way is more generalized. Table X provides a comparison of some of the advantages and disadvantages of these incentive schemes by classifying them based on their design principle.

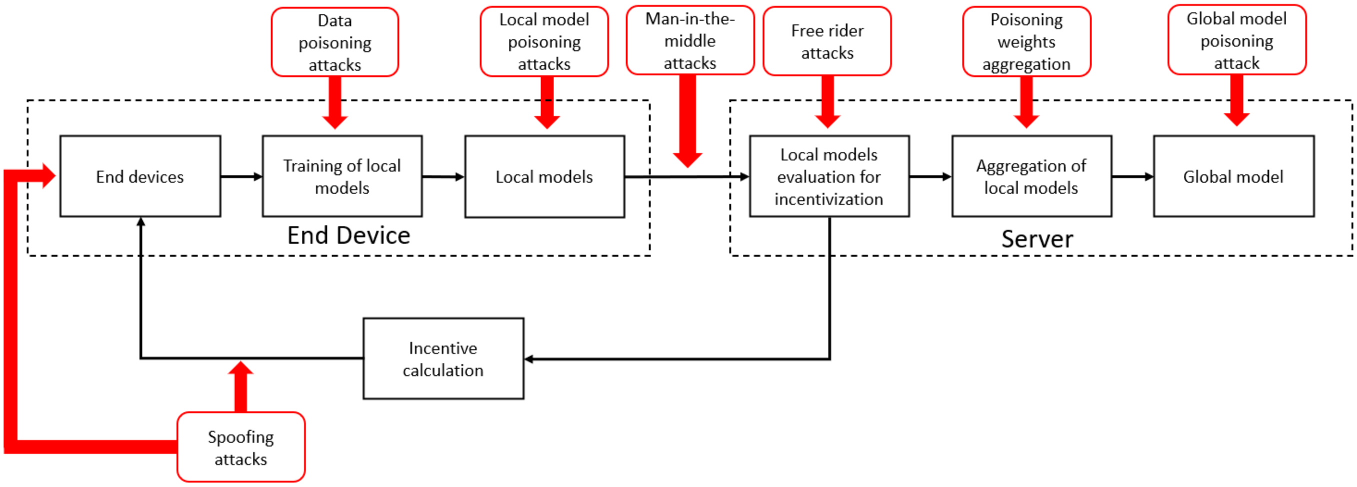
Data poisoning attacks
Local model poisoning attacks
Man-in-the-middle attacks
Free rider attacks
Poisoning weights aggregation
Global model poisoning attack
End devices
Training of local models
Local models
End Device
Local models evaluation for incentivization
Aggregation of local models
Global model
Server
Incentive calculation
Spoofing attacks

Fig. 12: The pipeline for incentive-driven federated learning (FL) depicting the vulnerabilities at different stages.

V. Security of Incentives Enabled FL (RQ2)

In this section, we first describe different attacks that can be realized in the FL settings. Then, we will provide the thematic analysis of eleven papers that are focused on the security aspects in the incentive-driven FL setting. As we did earlier, we classified these papers based on the design principle for incentive schemes and there are four categories: (i) Contract Theory, (ii) Game Theory, (iii) Deep Reinforcement Learning, and (iv) Blockchain.

A. Background Related to Security of FL

Despite the remarkable ability of FL to preserve data privacy, it still faces a major obstruction due to various security concerns, i.e., FL systems are susceptible to malicious attacks that hinder its practical implementation. As the server does not have access to the user's private data, the users can cheat or mislead the server by sending malicious updates to realize adversarial attacks (to get the intended outcomes). The adversarial attacks are generally classified into two types, i.e., untargeted attacks, and targeted attacks. The former deals with the attacks in which the goal of the adversary is to reduce the overall performance of the model. In the latter, the goal of the adversary is to modify the behavior of the model on some specific (selected) data instances while keeping the overall performance unaffected. In this paper, we first develop a pipeline of incentive-enabled FL by highlighting the various vulnerabilities at each stage (as shown in Figure 12). We then divided the attacks on FL into further types based on their nature.

1) Data Poisoning Attacks: In data poisoning attacks, the attacker adds malicious examples into the training data of a compromised ED. These kinds of attacks are realized at the training phase when the EDs are training their local models. Data poisoning attacks can either be targeted or untargeted. Data poisoning attacks have been largely studied in the literature [93]–[96] and to counter such attacks, several defense strategies have been proposed in the literature [21]–[23]. These works proposed different aggregation rules that improve the robustness of the models against data poisoning attacks. However, these rules are suitable for incentive-free FL systems, and for incentive-based FL, these defenses may defend only in cases where incentives are calculated after aggregation and not otherwise.

2) Local Model Poisoning Attacks: In local model poisoning attacks, the attacker manipulates the parameters of trained models and sends the malicious model to the MO [97]–[99]. FL is susceptible to such attacks after the computation of the local models as shown in Figure 12. To tackle model manipulation attacks, Fang et al. [97] developed two defenses–namely, error-based rejection and loss function-based rejection. Similarly, Li et al. [100] and Wu et al. [101] also presented defenses for model poisoning attacks.

3) Man-in-the-Middle Attacks: In man-in-the-middle attacks, the attacker manipulates the communication protocol between the server and the EDs to obtain sensitive information. The invader positions himself between the server and the EDs while they are communicating to listen and modify the traffic (data) between the two. The goal is to steal, spy, disrupt, or corrupt the data [102]. In the case of incentive-based FL, these attacks turn out to be very effective where an attacker can corrupt the data of other EDs to get more rewards for itself. Moreover, an attacker can also spy and disrupt the payment information to malfunction the FL system. This area is still unexplored and demands attention from the research community to develop effective attacks and defenses, especially for incentive-enabled FL environments.

4) Free Riding Attacks: In free-riding attacks, EDs (known as free riders) craft local model updates without using any training data [103]. The goal of free riders is to trick the system and get the rewards, hence affecting both the global model accuracy and the reward distribution. For instance, the authors in [103] developed two attacks: (i) Random generation of model parameters, and (ii) Delta weights attacks. In addition to these attacks, the authors also proposed a defense termed STD-DAGMM. Similarly, [104] proposed a robust and fair FL framework to defend against such attacks.

5) Poisoning Weights Aggregation: In this case, it is considered that the server is malicious and its goal is to obtain sensitive information from local models. In the FL pipeline, this type of attack usually occurs at the aggregation stage of local models at the server as shown in Figure 12. In some cases, in addition to servers, an ED may also try to steal sensitive information from other EDs. To deal with such attacks, secure aggregation protocols have been proposed in [9]. The assumption that the server is secure and honest is not deemed practical, therefore, the server can also become malicious. For incentive-based FL, many solutions employ blockchain which provides a robust and secure way to handle transactions as well as aggregation of local models.

6) Global Model Poisoning Attacks: In global model poisoning attacks, the malicious server/attacker manipulates the global model to disrupt the performance of the overall system. This attack occurs after the aggregation of the local models into a global model. These attacks can be performed by rival companies/MOs to fail other systems.

7) Spoofing Attacks: In spoofing attacks, the malicious player pretends itself as another device of the network and can attack the host, obtain information about the system, or spread malware in the network. In the context of FL, an attacker can pretend as an ED for entering the network and can steal model updates, upload fake updates or manipulate the updates of other EDs. In incentive-based FL, an attacker can steal the identity of other EDs and hence steal their rewards.

B. Analysis of FL Incentive Schemes from the Security Perspective

This section discusses the systematically selected incentive schemes that also considered security aspects. Table XI provides the thematic analysis of these incentive schemes by highlighting their goal, approach, attack type, and prevention technique(s). The details are described below.

1) Contract Theory-Based Mechanism: In real-life FL tasks contain several EDs, there might be unreliable workers that may try to mislead the global model training by intentionally performing in undesirable ways. As a solution, Kang et al. [14] proposed a reputation metric that can be used along with the incentives to separate reliable from unreliable EDs. Furthermore, they proposed contract theory-based incentivization to allure high reputation data owners to participate in the training process. In [115], a multi-weight subjective logic model managed by consortium blockchain is used to calculate the reputation of EDs. To evaluate the reliability of the local model updates, the system uses the Reject on Negative Influence (RONI) [116] scheme for Independently and Identically Distributed (IID) data scenario and FoolsGold [117] scheme for non-IID data scenario. Their proposed scheme also considers an unintentionally unreliable update sent by a reliable ED. They performed simulation on

TABLE XI: Incentive schemes accompanying security aspects.

Article	Target/Goal	Approach	Type of attack	Prevention technique	System setup	Dataset
Kang et al. [14]	Securing against unreliable EDs	Provide incentives based on the resources and reputation-based metric of the resources	Model poisoning attack	Two attack detection techniques, i.e., Reject on negative influence (RONI) for IID data scenario and FoolsGold for non-IID data scenario are used to detect unreliable data owners	20 Publishers 90 well-behaved EDs 10 unreliable EDs	MNIST
Richardson et al. [105]	Tackle the problem of security and data redundancy	Influence metric to determine the rewards	Free rider attack	The system utilizes the influence metric to detect the free riders.	Not mentioned	Not mentioned
Sun et al. [106]	Detect unreliable and malicious EDs	Incentives are provided based on participation level	Data poisoning attack	Reputation mechanism is used to exclude malicious participants from the training. If the reputation is less than a threshold, the ED loses the qualification to participate in training	1 Publisher 20 EDs at a time	MNIST
Wang et al. [107]	Reducing the privacy and security risk	Incentives are provided based on data quality and energy consumption	Man-in-the-middle attack	The system dynamically changes the public key of transactions between EDs and MO. Moreover, the hash-chained data structure property of blockchain address the problem of such attacks	1 Publisher 4 MEC nodes Uniformly distributed UAVs	MNIST
Feng et al. [108]	Enhancing security	Reputation metric is employed to distribute incentives among EDs	Data posioning attack	Reputation metric is used to filter out the malicious EDs	1 Publisher 30 EDs 3 MEC nodes	MNIST
Wang et al. [109]	Tackle the challenge of privacy and security	Incentives are provided based on the accuracy of global model	Data posioning attack	The systems utilizes check-point based smart contract to keep track of the attacker	Not mentioned	Not mentioned
Weng et al. [110]	Guarantees data privacy for EDs and auditability for the whole training process	EDs get rewards based on their contribution	Data poisoning attack	The system utilizes smart contract to regulate the behavior and track the attackers	1 Publisher 4-10 EDs	MNIST
Bao et al. [111]	Introduce auditability of malicious EDs and reduce dishonest cooperation	The price of the models are set according to expectation of EDs	Data poisoning attack	A CCM is used to evaluate the reliability of local updates	1 Publisher 4 & 10 EDs	SHA-512 MHT
Zhao et al. [112]	Enhancing data privacy and protection against malicious attacks	The incentive provided to the EDs is proportional to their reputation	Model poisoning attack	Multi-Krum is used to detect and filter out malicious participants	1 Publisher 10 EDs	MNIST
Li et al. [113]	Securing against malicious clients	Rewards are distributed among EDs based on their score	Data poisoning attack	A CCM is used to verify the integrity of local model updates	1 Publisher 900 EDs	FEMNIST
Qu et al. [114]	To enhance the accuracy and robustness	Reward is proportional to data size	Data poisoning attack	A consensus mechanism is used to tackle the problem of poisoning attacks	Not mentioned	CIFAR-10

MNIST dataset with 20 task publishers, 90 well-behaved EDs, and 10 unreliable EDs. Their results show that their proposed technique outperforms the Stackelberg game-based techniques (proposed in [56]–[58]).

2) Game Theory-Based Mechanisms: Incentives are provided to EDs to motivate them to provide truthful data. Game-theoretic schemes are built to reward the EDs on the basis of this truthful data, however, these schemes do not consider data redundancy with previous contributions. This data redundancy creates an opportunity for EDs to trick the MO and earn more rewards, i.e., EDs are enticed to free-riding attacks. To solve this problem, Richardson et al. [105] designed a mechanism based on an influence metric that guarantees truthful reporting is the best strategy for EDs. Moreover, they also provide a bound on the incentive budget, i.e., the budget must be in proportion to the value of the FL model. Influence is an excellent measure since it rewards the contributors that contribute significantly to obtain the desired accuracy as soon as possible. They modeled the interaction between the server and EDs as a game where EDs design their strategies based on rewards. The influence is calculated by taking the difference in loss function between the model trained with and without the concerned data and the EDs are rewarded in proportion to this metric. This sounds impractical being communication and computation-intensive because it means that each FL round is repeated N times assuming there are N EDs so that in each of those rounds (or sub rounds) one ED is excluded to see the effect of training without that specific ED.

Sun et al. [106] designed a dynamic digital twin architecture for the air-ground network where the drone works as the aggregator and the ground clients as the EDs. In their proposed mechanism, the interaction between the FL entities is considered using the Stackelberg game. The aggregator acts as a leader and sets the preferences for EDs and EDs act as the followers and decide their training strategies. To make the system more accurate, a contribution measurement mechanism was proposed which selects a set of appropriate EDs to participate in the training. A reputation metric is employed to detect unreliable and malicious EDs. Firstly, the authors designed a static incentive scheme where they considered a limited scenario of a small-scale air-ground network where all the EDs are within the range of a single drone. The drone publishes the reward rate along with the FL task and EDs determine the participation level based on reward and training cost. Furthermore, they proposed a dynamic incentive scheme for handling high mobility large-scale networks where a single drone is not able to cover the entire area. In the dynamic incentive scheme, the set of clients can change dynamically depending upon the position of the drone. In the static scheme, the clients decide the number of rounds that it participates in the global updates while in dynamic, the clients decide the number of local training rounds. They showed that the dynamic scheme outperforms the static and the benchmark scheme. In the proposed scheme, they assumed that the aggregator and EDs know the training cost and reputation value of all the EDs which is not a realistic assumption.

Synthesis and reflection: Truthful reporting is not always the most suitable strategy for EDs as assumed in most of the works. Therefore, the above articles introduce some metrics to monitor the behavior of the EDs in the FL training. These metrics force the EDs to truthfully report their type and contributed resources. These metrics are also able to detect if some EDs are behaving maliciously or not.

3) Deep Reinforcement Learning Based Mechanisms: Security and privacy risks exist in the mobile crowd-sensing of UAVs due to vulnerabilities of the central server and unreliable EDs. Wang et al. [107] developed SFAC, a secure FL framework for UAV-assisted mobile crowd-sensing. They employed blockchain and differential privacy to securely exchange the local model updates while preserving the privacy of shared models. To motivate the high-quality UAVs to participate in the training process, they adopt a two-tier RL-based incentive mechanism. They showed that the proposed scheme motivates the high-quality UAVs to participate at a low cost as compared to others.

4) Blockchain Based Mechanisms: Feng et al. [108] proposed the usage of blockchain and reputation to deal with security issues. They proposed a two-layer architecture that consists of two types of blockchain: local model update chain and global model update chain. The former is designed to efficiently keep track of all local model updates and to build a reputation system. While the latter serves the purpose of securing the proposed system from malfunctioning and divide it into logically isolated task-specific chains. To develop the incentive mechanism, they employed smart contracts and proposed a reputation-learning-based scheme. They considered two types of rewards, FL reward, and blockchain reward. FL reward deals with providing incentives to EDs based on the data size, data quality, and satisfaction of the task. However, this quantitative criterion is unknown amid information asymmetry. To deal with such discrepancies, they make use of reputation and DL to determine the FL rewards. Blockchain rewards deal with paying miners due to their services for the verification and generating blocks. The results indicate that the accuracy achieved by the proposed system is higher than FLchain [111].

Taking into account the challenge of privacy and security, Wang et al. [109] designed a system BlockFedML which prevents the information leakage from model updates and also protects against integrity attacks. They exploit blockchain using smart contracts to design an incentive mechanism. The purpose of incentivization is to encourage ED's to contribute high-quality data that can improve the accuracy of the model. The computing node follows the smart contract to determine its contribution and then selects a training round to participate. If the accuracy of the model is improved, the node will be rewarded otherwise it will be penalized. The system utilizes checkpoint-based smart contracts to keep track of the attacker. The checkpoint-based smart contract provides the flexibility to recover the model to its previous state if an attacker is detected. The proposed system checks the accuracy of the global model using each local update and if accuracy decreases, that ED is considered malicious.

Weng et al. [110] proposed DeepChain, an auditable and privacy-preserving DL with blockchain-based incentive. Their setup consists of two entities: (i) data owners who do not have sufficient data and resources to train the model alone and have to collaborate with each other to train the model forming a party; and (ii) miners who process transactions for model updates and create blocks on DeepChain. The data owners launch transactions and pay a fee for a transaction depending upon the amount of data owned by it—i.e., the more the data is available, the less will be the fee. The miners compete with each other to process the transaction. The winning miner becomes the leader and creates a new block on DeepChain and gets the reward. If a party poses an invalid transaction and the miner processes it, they both will be penalized. Due to the value-based incentive, the participants are forced to behave correctly during the process. The system utilizes smart contracts to regulate the behavior and track the attackers. The system checks the accuracy of the global model using each local update and if the accuracy decreases, that ED is considered malicious.

Bao et al. [111] argued that previous blockchain techniques for FL only focus on ML-related parameters and ignore the contribution and reliability of the EDs. As a solution, they proposed a scheme named *FLChain* to introduce the auditability of malicious trainers and to reduce dishonest cooperation. For fair profit partition, the reliability and contribution of data owners are determined, and then an algorithm is designed for maximizing the total profit of data owners. They proposed to replace the traditional FL server with FLChain which is an FL server secured by blockchain. Data owners are selected by FLChain depending upon their collaborating intention and reliability. After the global model aggregation, data owners decrypt the aggregated model and upgrade their local models. Each encrypted model and decrypted share can be inspected by its proof and the dishonest workers will be reported and punished. After the completion of training, the model will be purchased by FLChain. The price of that model is set according to the expectation of each data owner. Dishonest trained models are not allowed for sale and the dishonesty detectors are rewarded for successful detection and affected data owners are compensated.

Zhao et al. [112] employed blockchain to design an incentive scheme for FL in IoT with a focus on data privacy and preventing malicious attacks. EDs which fulfill the task requirements, begin to train the model locally, add the private keys, and upload the trained model to the blockchain. If the miner decides the signature with a private key to be invalid, it rejects the update. A group of selected miners known as verifiers, calculate the reputation using Multi-KRUM approach [118]. Based on the reputation, the hostile updates are filtered out and the incentive is provided to EDs, which is proportional to their reputation. To enforce privacy, they proposed to introduce Laplacian noise during training in the batch normalization layer. Furthermore, they discussed that their proposed scheme might bring delays in applications involving a large number of EDs. They argued that this problem can be solved by providing incentives to users based on the training time like the one proposed in [62].

Li et al. [113] devised a system using Committee Consensus Mechanism (CCM) [119] to incentivize the FL process while securing against malicious attacks. In this framework, the central server is replaced by blockchain which controls the data owners and supports global and local models. A CCM is used to verify the local updates, i.e, to check whether the updates are from honest EDs or not, before adding them to the chain. In CCM, a few honest data owners are elected to constitute a committee whose task is to verify local updates and block generation on-chain. The participants train the local models and send them to a committee which then validates them. To enter the FL training process, each data owner has to pay a permission fee to access the global models and this fee is kept by committee members. After aggregation of the local updates, committee members distribute the reward among the participants based on their scores.

Qu et al. [114] tackled three main problems, i.e., low efficiency, poisoning attacks, and incentivization, of data-driven cognitive computing in the industry 4.0 settings. They employed blockchain to develop a decentralized cognitive computing (D2C) model in the FL paradigm. To provide guarantees against poisoning attacks and fair incentivization, they use the proof-of-work (PoW) consensus algorithm. They consider two types of incentive mechanisms, i.e., data reward, and token reward. The data reward is provided to industry 4.0 machines for the contribution of their data and is proportional to data size. The token reward is provided to miners based on the aggregation provided by them. To further enhance the security and accuracy, they make use of Markov decision process for the selection of aggregators. They considered only the IID settings for evaluation which is not always the case.

Synthesis and reflections: Blockchain-based solutions employ certain techniques (such as smart contracts, checkpoint-based smart contracts, and Multi-KRUM, etc.) to determine the malicious and unreliable EDs. The consensus mechanism of blockchain provides robustness against certain types of attacks.

VI. Limitations and pitfalls (RQ3)

From our analysis of the systematically reviewed literature, it is evident that incentivization in FL attracts high-quality data owners that can collaborate to increase the efficiency of the FL models. It also discourages erroneous updates to the central model by imposing a penalty on the data owners that attempt to delude the central model. Furthermore, the use of incentivization helps in bringing trust, accountability, auditability, and integrity to the system. Our literature review reveals that a few reward schemes focus only on the maximization of profit for MOs; some are focused only on the profit of data owners; while some on the profit of both the MO and the data owners. The schemes focusing on maximizing the profit of both the MO and the data owners tend to have FL models with overall higher accuracy and can therefore be considered as more efficient.

Each design principle provides certain advantages and disadvantages of its own and there is a trade-off that needs to be considered according to the problem at hand. Contact theory-based methods are better at handling the information symmetry but they are more focused on optimizing the rewards of MO. Similarly, game theory-based mechanisms provide better optimization of resources and fair distribution of rewards but these schemes perform poorly amid information asymmetry. Hence, the design of incentive schemes is highly dependent on the problem being solved.

Most of the reward mechanisms considered that all participating data owners to be honest and they do not provide any malicious updates but a few of them also consider otherwise. A number of the articles proposed solutions that are very specific to the task at hand and can not be applied to other problems and there is a need for further research to make those solutions more generalized. Moreover, most of the research articles assumed that the data is IID across different users [120]. While in real situations, the data at the users' end might be imbalanced, and incomplete, i.e., some features or the samples of an entire class might be completely missing at certain EDs [121]. This leads to a highly imperfect local model which might mislead the global model when aggregated. There is a need for mechanisms that can ensure the sanctity of the local model parameters before being aggregated to the global model. As a solution, a few articles propose the usage of reputation metrics to differentiate between the data owners that intentionally upload malicious models and those who upload such incorrect models unintentionally.

Furthermore, FL systems are susceptible to data-poisoning and model-poisoning attacks. An adversary might get access to a single system or might even pose as a legitimate user in the FL system. In the former case, the adversary can poison the training data of the system. In the latter case, the adversary can upload tampered or poisoned model updates to the central entity. Out of 46 articles, only 11 considered the presence of adversaries in the systems, even though ensuring the security of FL systems is critical for the practical success of FL in the real world (where the presence of malicious actors is common).

VII. Open Research Issues

This section is related to our last research objective (O4) and in this section, we will elaborate upon some open research problems that require further development.

A. Investigating Unexplored Design Mechanisms

In recent years, substantial attention has been devoted to the development of incentive schemes for FL leveraging techniques from contract theory, game theory, auction theory, reputation-based approach, and blockchain.

However, there are certain design mechanisms of economics and computer science—such as bargaining games, trust and reputation, lotteries, and market-driven mechanisms—that have not been profitably explored in the field of incentivizing FL. These mechanisms can be quite useful for the design of the FL incentive schemes due to advantages such as those listed below.

- Bargaining games can benefit both the players, i.e., MO and EDs can propose offers to each other and they interact with each other to determine a price that benefits both MO and EDs. While in the case of contract theory, MO proposes the contract, and EDs either have to accept the offer or not.
- Trust and reputation-based approaches are useful when the FL environment is *Byzantine*, i.e., it contains malicious EDs. In addition, it is also useful when the FL system deals with non-monetary rewards. However, determination of the trust at the initial phase remains a major challenge.
- Lottery or Tullock Contest, a game in which a winner is determined by the probability of contest success function [122], is useful when we are concerned with mass participation of EDs in the FL system to increase the quantity of data.
- Market-driven mechanisms are only applicable where the FL system is treated as a market commodity or the user needs to perform market analysis of the FL system. Such mechanisms have been shown to be very useful in providing financial sustainability [123]. However, constant payment to EDs can cause a severe burden on the budget in the case of monetary incentives.

B. Designing Practical Reward Method

In the literature, the widely used reward methods are monetary rewards and non-monetary rewards. In the monetary reward method, EDs are given money based on their contribution whereas, in the non-monetary reward approach, there can be multiple ways to pay the EDs such as in the form of digital currency (e.g., cryptocurrency), tokens, points, and services. The literature of incentivized FL lacks discussion of reward type, i.e., the majority of articles do not clearly describe how the reward is paid to EDs. Therefore, the design of practical reward methods that are more suitable for real-world applications is required.

C. Adversarially Robust FL

The literature suggests that DL models are highly susceptible to carefully crafted adversarial perturbations, i.e., the adversarial ML attacks [124]–[126]. This vulnerability of ML/DL models can be exploited by an internal malicious ED or external adversary. The presence of such malicious actors is more common in FL due to its distributed nature and incentivization further provides a substantial motivation to such adversaries. In the literature, different attacks have been proposed for cloud-hosted ML models [127] and for the models trained using FL settings [98]. Considerable attention has been devoted to the development of mitigation strategies for adversarial attacks, however, the literature focused on attacking ML/DL-based systems is drastically increasing as compared with the literature focused on developing defense methods. Therefore, the development of adversarially robust ML/DL models stills remains an open research problem. On the other hand, the critical nature of the FL system demands a secure and robust operation of ML/DL systems.

D. Robustness to Data Imperfections and Drifts

The performance of a model being trained in an FL setting is highly dependent upon the quality of data owned by the EDs. The real-world data is likely to be incomplete, biased, and imbalanced. As the data collection in realistic settings is temporal and dynamic. This statistical heterogeneity significantly affects the performance of the global (shared) model. As a result, the convergence of the FL system becomes challenging. It demands the development of personalized approaches that could handle such imperfections. Another problem encountered in real-world data is the distributional shifts, where the test data diverges from the data used in training. The literature suggests that these distribution shifts lead to adversarial attacks [128], [129]. To withstand the integrity attacks, it is crucial to make ML/DL models robust to these distributional shifts.

VIII. Conclusions

The enhanced data privacy promised by Federated Learning (FL) makes it attractive to researchers and practitioners but incentivization is necessary for the successful survival of the FL community. In this paper, we presented a systematic review of literature related to incentivization in FL and the associated security challenges. The relevant articles were collected from five major publishers that include IEEE Xplore, ACM Digital Library, Elsevier, Springer, and arXiv. We develop a review mechanism based on inclusion and exclusion criteria for the selection of relevant articles. The selected papers were then analyzed and categorized into seven themes. This paper will provide a foundation for researchers to understand the process of incentivization and to get insights for designing new incentive mechanisms. We, then, discussed the security challenges associated with incentive-driven FL. Furthermore, we provide a detailed discussion on the strengths and weaknesses of the existing incentive schemes. We also highlight the open research issues that will enable the researchers to investigate further.

Acknowledgement

AQ: This publication was made possible by NPRP grant number 13S-0206-200273 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors.

References

[1] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Transactions on Intelligent Systems and Technology*, vol. 10, No. 2, 2019. [Online]. Available: http://arxiv.org/abs/1902.04885

[2] J. Konecny, H. B. McMahan, D. Ramage, and P. Richtarik, "Federated optimization: Distributed machine learning for on-device intelligence," *CoRR*, vol. abs/1610.02527, 2016. [Online]. Available: http://arxiv.org/abs/1610.02527

[3] A. Hard, K. Rao, R. Mathews, S. Ramaswamy, F. Beaufays, S. Augenstein, H. Eichner, C. Kiddon, and D. Ramage, "Federated learning for mobile keyboard prediction," *arXiv preprint arXiv:1811.03604*, 2018.

[4] N. Rieke, J. Hancox, W. Li, F. Milletarì, H. R. Roth, S. Albarqouni, S. Bakas, M. N. Galtier, B. A. Landman, K. Maier-Hein, S. Ourselin, M. Sheller, R. M. Summers, A. Trask, D. Xu, M. Baust, and M. J. Cardoso, "The future of digital health with federated learning," *npj Digital Medicine*, vol. 3,119, 2020.

[5] A. Qayyum, K. Ahmad, M. A. Ahsan, A. Al-Fuqaha, and J. Qadir, "Collaborative federated learning for healthcare: Multi-modal covid-19 diagnosis at the edge," *arXiv preprint arXiv:2101.07511*, 2021.

[6] L. U. Khan, W. Saad, Z. Han, E. Hossain, and C. S. Hong, "Federated learning for internet of things: Recent advances, taxonomy, and open challenges," *arXiv preprint arXiv:2009.13012*, 2020.

[7] Z. Du, C. Wu, T. Yoshinaga, K. L. A. Yau, Y. Ji, and J. Li, "Federated learning for vehicular internet of things: Recent advances and open issues," *IEEE Open Journal of the Computer Society*, vol. 1, pp. 45–61, 2020.

[8] S. Elnagar and M. Thomas, "Federated deep learning: A conceptual model and applied framework for industry 4.0," in *Americas Conference on Information Systems (AMCIS)*, 2020.

[9] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth, "Practical secure aggregation for privacy-preserving machine learning," in *CCS'17*. New York, NY, USA: Association for Computing Machinery, 2017. [Online]. Available: https://doi.org/10.1145/3133956.3133982

[10] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Processing Magazine*, vol. 37, no. 3, p. 50–60, May 2020. [Online]. Available: http://dx.doi.org/10.1109/MSP.2020.2975749

[11] S. A. Rahman, H. Tout, H. Ould-Slimane, A. Mourad, C. Talhi, and M. Guizani, "A survey on federated learning: The journey from centralized to distributed on-site learning and beyond," *IEEE Internet of Things Journal*, pp. 1–1, 2020.

[12] Y. Liu, L. Zhang, N. Ge, and G. Li, "A systematic literature review on federated learning: From a model quality perspective," *arXiv preprint arXiv:2012.01973*, 2020.

[13] P. Kairouz *et al.*, "Advances and open problems in federated learning," *arXiv preprint arXiv:1912.04977*, 2019.

[14] J. Kang, Z. Xiong, D. Niyato, S. Xie, and J. Zhang, "Incentive mechanism for reliable federated learning: A joint optimization approach to combining reputation and contract theory," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10 700–10 714, 2019.

[15] L. U. Khan, S. R. Pandey, N. H. Tran, W. Saad, Z. Han, M. N. H. Nguyen, and C. S. Hong, "Federated learning for edge networks: Resource optimization and incentive mechanism," *IEEE Communications Magazine*, vol. 58, no. 10, pp. 88–93, 2020.

[16] Y. Zhan, P. Li, S. Guo, and Z. Qu, "Incentive mechanism design for federated learning: Challenges and opportunities," *IEEE Network*, 2021.

[17] Y. Zhan, J. Zhang, Z. Hong, L. Wu, P. Li, and S. Guo, "A survey of incentive mechanism design for federated learning," *IEEE Transactions on Emerging Topics in Computing*, 2021.

[18] Q. Li, Z. Wen, Z. Wu, S. Hu, N. Wang, Y. Li, X. Liu, and B. He, "A survey on federated learning systems: vision, hype and reality for data privacy and protection," *arXiv preprint arXiv:1907.09693*, 2019.

[19] 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.

[20] J. Kang, Z. Xiong, D. Niyato, H. Yu, Y. Liang, and D. I. Kim, "Incentive design for efficient federated learning in mobile networks: A contract theory approach," in *2019 IEEE VTS Asia Pacific Wireless Communications Symposium (APWCS)*, 2019, pp. 1–5.

[21] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer, "Machine learning with adversaries: Byzantine tolerant gradient descent," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, pp. 118–128.

[22] E. M. E. Mhamdi, R. Guerraoui, and S. Rouault, "The hidden vulnerability of distributed learning in byzantium," *arXiv preprint arXiv:1802.07927*, 2018.

[23] D. Yin, Y. Chen, R. Kannan, and P. Bartlett, "Byzantine-robust distributed learning: Towards optimal statistical rates," in *International Conference on Machine Learning*. PMLR, 2018, pp. 5650–5659.

[24] J. Konecny, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," *arXiv preprint arXiv:1610.05492*, 2017.

[25] Q. Yang, Y. Liu, Y. Cheng, Y. Kang, T. Chen, and H. Yu, "Federated learning," *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 13, no. 3, pp. 1–207, 2019.

[26] P. Bolton, M. Dewatripont *et al.*, *Contract theory*. MIT press, 2005.

[27] Y. Zhang, M. Pan, L. Song, Z. Dawy, and Z. Han, "A survey of contract theory-based incentive mechanism design in wireless networks," *IEEE Wireless Communications*, vol. 24, no. 3, pp. 80–85, 2017.

[28] N. Gupta, S. K. Dhurandher, and B. Kumar, "Contract theory based incentive mechanism design approaches in cognitive radio networks: A survey," in *2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU)*, 2019, pp. 1–6.

[29] L. Kockesen and E. A. Ok, *An Introduction to Game Theory*. New York University, 2007.

[30] E. Rasmusen, *Games and Information: An Introduction to Game Theory*. Wiley-Blackwell, 2006.

[31] R. Jia, D. Dao, B. Wang, F. A. Hubis, N. Hynes, N. M. Gurel, B. Li, C. Zhang, D. Song, and C. J. Spanos, "Towards efficient data valuation based on the shapley value," in *Proceedings of Machine Learning Research*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and M. Sugiyama, Eds., vol. 89. PMLR, 16–18 Apr 2019, pp. 1167–1176. [Online]. Available: http://proceedings.mlr.press/v89/jia19a.html

[32] M. O. Jackson, "Mechanism theory," *Optimization and Operations Research in the Encyclopedia of Life support Systems, EOLSS*, pp. 228–277, 2003.

[33] T. Borgers, D. Krahmer, and R. Strausz, *An Introduction to the Theory of Mechanism Design*. Oxford University Press, 2015.

[34] R. B. Myerson, "Perspectives on mechanism design in economic theory," *American Economic Review*, vol. 98, no. 3, pp. 586–603, 2008. [Online]. Available: http://www.aeaweb.org/articles.php?doi=10.1257/aer.98.3.586

[35] T. S. Chandrashekar, Y. Narahari, C. H. Rosa, D. M. Kulkarni, J. D. Tew, and P. Dayama, "Auction-based mechanisms for electronic procurement," *IEEE Transactions on Automation Science and Engineering*, vol. 4, no. 3, pp. 297–321, 2007.

[36] B. Eraslan, D. Gozupek, and F. Alagoz, "An auction theory based algorithm for throughput maximizing scheduling in centralized cognitive radio networks," *IEEE Communications Letters*, vol. 15, no. 7, pp. 734–736, 2011.

[37] P. Klemperer, *Auctions: theory and practice*. Princeton University Press, 2004.

[38] S. Nakamoto and A. Bitcoin, "A peer-to-peer electronic cash system," *Bitcoin.–URL: https://bitcoin. org/bitcoin. pdf*, vol. 4, 2008.

[39] Z. Zhang, S. Xie, H.-N. Dai, X. Chen, and H. Wang, "Blockchain challenges and opportunities: A survey," *Int. J. Web Grid Serv.*, vol. 14, no. 4, p. 352–375, Jan. 2018.

[40] G. Peters and E. Panayi, "Understanding modern banking ledgers through blockchain technologies: Future of transaction processing and smart contracts on the internet of money," *SSRN*, 2015. [Online]. Available: http://dx.doi.org/10.2139/ssrn.2692487

[41] A. L. Tsilidou and G. Foroglou, "Further applications of blockchain," *12th Student Conference on Managerial Science and Technology*, 2015.

[42] V. François-Lavet, P. Henderson, R. Islam, M. G. Bellemare, and J. Pineau, "An introduction to deep reinforcement learning," *Foundations and Trends in Machine Learning*, vol. 11, no. 3-4, 2018.

[43] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[44] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep reinforcement learning: A brief survey," *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 26–38, 2017.

[45] N. C. Luong, D. T. Hoang, S. Gong, D. Niyato, P. Wang, Y.-C. Liang, and D. I. Kim, "Applications of deep reinforcement learning in communications and networking: A survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3133–3174, 2019.

[46] H. H. Zhuo, W. Feng, Q. Xu, Q. Yang, and Y. Lin, "Federated reinforcement learning," *arXiv preprint arXiv:1901.08277*, 2019.

[47] N. Ding, Z. Fang, and J. Huang, "Incentive mechanism design for federated learning with multi-dimensional private information," in *2020 18th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOPT)*, 2020, pp. 1–8.

[48] W. Y. B. Lim, J. Huang, Z. Xiong, J. Kang, D. Niyato, X.-S. Hua, C. Leung, and C. Miao, "Towards federated learning in uav-enabled internet of vehicles: A multi-dimensional contract-matching approach," *arXiv preprint arXiv:2004.03877*, 2020.

[49] W. Y. B. Lim, Z. Xiong, C. Miao, D. Niyato, Q. Yang, C. Leung, and H. V. Poor, "Hierarchical incentive mechanism design for federated machine learning in mobile networks," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 9575–9588, 2020.

[50] W. Y. B. Lim, Z. Xiong, J. Kang, D. Niyato, C. S. Leung, C. Miao, and S. Shen, "When information freshness meets service latency in federated learning: A task-aware incentive scheme for smart industries," *IEEE Transactions on Industrial Informatics*, 2020.

[51] W. Y. B. Lim, Z. Xiong, J. Kang, D. Niyato, Y. Zhang, C. Leung, and C. Miao, "An incentive scheme for federated learning in the sky," in *Proceedings of the 2nd ACM MobiCom Workshop on Drone Assisted Wireless Communications for 5G and Beyond*, 2020, pp. 55–60.

[52] W. Y. B. Lim, Z. Xiong, D. Niyato, J. Huang, X.-S. Hua, and C. Miao, "Incentive mechanism design for federated learning in the internet of vehicles," in *2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall)*. IEEE, 2020, pp. 1–5.

[53] K. L. Ng, Z. Chen, Z. Liu, H. Yu, Y. Liu, and Q. Yang, "A multi-player game for studying federated learning incentive schemes," in *Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI 2020)*, 2020, pp. 5179–5281.

[54] C. Ilias and S. Georgios, "Machine learning for all: A more robust federated learning framework," in *Proc. 5th Int. Conf. Inf. Syst. Secur. Privacy*, 2019, pp. 544–551.

[55] J. Domingo-Ferrer, A. Blanco-Justicia, D. Sánchez, and N. Jebreel, "Co-utile peer-to-peer decentralized computing," in *2020 20th IEEE/ACM International Symposium on Cluster, Cloud and Internet Computing (CCGRID)*. IEEE, 2020, pp. 31–40.

[56] J. Kang, Z. Xiong, D. Niyato, D. Ye, D. I. Kim, and J. Zhao, "Toward secure blockchain-enabled internet of vehicles: Optimizing consensus management using reputation and contract theory," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 2906–2920, 2019.

[57] Z. Hou, H. Chen, Y. Li, and B. Vucetic, "Incentive mechanism design for wireless energy harvesting-based internet of things," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2620–2632, 2018.

[58] T. Liu, J. Li, F. Shu, M. Tao, W. Chen, and Z. Han, "Design of contract-based trading mechanism for a small-cell caching system," *IEEE Transactions on Wireless Communications*, vol. 16, no. 10, pp. 6602–6617, 2017.

[59] Y. Sarikaya and O. Ercetin, "Motivating workers in federated learning: A stackelberg game perspective," *IEEE Networking Letters*, vol. 2, no. 1, pp. 23–27, 2020.

[60] S. Feng, D. Niyato, P. Wang, D. I. Kim, and Y. Liang, "Joint service pricing and cooperative relay communication for federated learning," in *2019 International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 2019, pp. 815–820.

[61] S. R. Pandey, N. H. Tran, M. Bennis, Y. K. Tun, A. Manzoor, and C. S. Hong, "A crowdsourcing framework for on-device federated learning," *IEEE Transactions on Wireless Communications*, vol. 19, no. 5, pp. 3241–3256, 2020.

[62] H. Yu, Z. Liu, Y. Liu, T. Chen, M. Cong, X. Weng, D. Niyato, and Q. Yang, "A sustainable incentive scheme for federated learning," *IEEE Intelligent Systems*, vol. 35, no. 04, pp. 58–69, jul 2020.

[63] R. Hu and Y. Gong, "Trading data for learning: Incentive mechanism for on-device federated learning," *arXiv preprint arXiv:2009.05604*, 2020.

[64] X. Qu, Q. Hu, and S. Wang, "Privacy-preserving model training architecture for intelligent edge computing," *Computer Communications*, vol. 162, pp. 94 – 101, 2020. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0140366420318582

[65] J. Lee, D. Kim, and D. Niyato, "Market analysis of distributed learning resource management for internet of things: a game-theoretic approach," *IEEE Internet of Things Journal*, vol. 7, no. 9, pp. 8430–8439, 2020.

[66] Y. Sarikaya and O. Ercetin, "Regulating workers in federated learning by yardstick competition," in *Proceedings of the 13th EAI International Conference on Performance Evaluation Methodologies and Tools*, 2020, pp. 150–155.

[67] T. Song, Y. Tong, and S. Wei, "Profit allocation for federated learning," in *2019 IEEE International Conference on Big Data (Big Data)*. IEEE, 2019, pp. 2577–2586.

[68] C. T. Dinh, N. H. Tran, M. N. Nguyen, C. S. Hong, W. Bao, A. Y. Zomaya, and V. Gramoli, "Federated learning over wireless networks: Convergence analysis and resource allocation," *IEEE/ACM Transactions on Networking*, 2020.

[69] Y. Zhan, P. Li, Z. Qu, D. Zeng, and S. Guo, "A learning-based incentive mechanism for federated learning," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6360–6368, 2020.

[70] Y. Zhan and J. Zhang, "An incentive mechanism design for efficient edge learning by deep reinforcement learning approach," in *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications*, 2020, pp. 2489–2498.

[71] E. Roth, *The Shapley Value: Essays in Honor of Lloyd S. Shapley*. Cambridge University Press, 1988, pp. i–iv.

[72] Y. Liu, S. Sun, Z. Ai, S. Zhang, Z. Liu, and H. Yu, "Fedcoin: A peer-to-peer payment system for federated learning," *arXiv preprint arXiv:2002.11711*, 2020.

[73] M. Cong, H. Yu, X. Weng, J. Qu, Y. Liu, and S. M. Yiu, "A vcg-based fair incentive mechanism for federated learning," *arXiv preprint arXiv:2008.06680*, 2020.

[74] S. Kim, “Incentive design and differential privacy based federated learning: A mechanism design perspective,” *IEEE Access*, vol. 8, pp. 187 317–187 325, 2020.

[75] M. Cong, H. Yu, X. Weng, and S. M. Yiu, “A game-theoretic framework for incentive mechanism design in federated learning,” in *Federated Learning*. Springer, 2020, pp. 205–222.

[76] N. Nisan and A. Ronen, “Computationally feasible vcg mechanisms,” *J. Artif. Int. Res.*, vol. 29, no. 1, p. 19–47, May 2007.

[77] T. H. T. Le, N. H. Tran, Y. K. Tun, M. N. H. Nguyen, S. R. Pandey, Z. Han, and C. S. Hong, “An incentive mechanism for federated learning in wireless cellular network: An auction approach,” *arXiv preprint arXiv:2009.10269*, 2020.

[78] R. Zeng, S. Zhang, J. Wang, and X. Chu, “Fmore: An incentive scheme of multi-dimensional auction for federated learning in mec,” *arXiv preprint arXiv:2002.09699*, 2020.

[79] Y. Jiao, P. Wang, D. Niyato, B. Lin, and D. I. Kim, “Toward an automated auction framework for wireless federated learning services market,” *arXiv preprint arXiv:1912.06370*, 2020.

[80] T. H. T. Le, N. H. Tran, Y. K. Tun, Z. Han, and C. S. Hong, “Auction based incentive design for efficient federated learning in cellular wireless networks,” in *2020 IEEE Wireless Communications and Networking Conference (WCNC)*, 2020, pp. 1–6.

[81] C. Ying, H. Jin, X. Wang, and Y. Luo, “Double insurance: Incentivized federated learning with differential privacy in mobile crowdsensing,” in *2020 International Symposium on Reliable Distributed Systems (SRDS)*. IEEE, 2020, pp. 81–90.

[82] P. Roy, S. Sarker, M. A. Razzaque, M. Mamun-or Rashid, M. M. Hassan, and G. Fortino, “Distributed task allocation in mobile device cloud exploiting federated learning and subjective logic,” *Journal of Systems Architecture*, p. 101972, 2020.

[83] S. Arora and B. Barak, *Computational complexity: a modern approach*. Cambridge Univ. Press, 2010.

[84] X. Zhou, Z. Zhang, G. Wang, X. Yu, B. Y. Zhao, and H. Zheng, “Practical conflict graphs in the wild,” *IEEE/ACM Transactions on Networking*, vol. 23, no. 3, pp. 824–835, 2015.

[85] C. J. C. H. Watkins and P. Dayan, “Q-learning,” in *Machine Learning*, 1992, pp. 279–292.

[86] K. Toyoda and A. N. Zhang, “Mechanism design for an incentive-aware blockchain-enabled federated learning platform,” in *2019 IEEE International Conference on Big Data (Big Data)*, 2019, pp. 395–403.

[87] W. Zhang, Q. Lu, Q. Yu, Z. Li, Y. Liu, S. K. Lo, S. Chen, X. Xu, and L. Zhu, “Blockchain-based federated learning for device failure detection in industrial IoT,” *IEEE Internet of Things Journal*, pp. 1–1, 2020.

[88] N. Q. Hieu, T. T. Anh, N. C. Luong, D. Niyato, D. I. Kim, and E. Elmroth, “Resource management for blockchain-enabled federated learning: A deep reinforcement learning approach,” *arXiv preprint arXiv:2004.04104*, 2020.

[89] S. Kumar, S. Dutta, S. Chatturvedi, and M. Bhatia, “Strategies for enhancing training and privacy in blockchain enabled federated learning,” in *2020 IEEE Sixth International Conference on Multimedia Big Data (BigMM)*. Los Alamitos, CA, USA: IEEE Computer Society, sep 2020, pp. 333–340. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/BigMM50055.2020.00058

[90] M. H. ur Rehman, K. Salah, E. Damiani, and D. Svetinovic, “Towards blockchain-based reputation-aware federated learning,” in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2020, pp. 183–188.

[91] V. Buterin, “A next-generation smart contract and decentralized application platform,” *Ethereum white paper*, 2014.

[92] X. Wu, Z. Liang, and J. Wang, “FedMed: A federated learning framework for language modeling,” *Sensors*, vol. 20, no. 14, p. 4048, 2020.

[93] V. Tolpegin, S. Truex, M. E. Gursoy, and L. Liu, “Data poisoning attacks against federated learning systems,” in *European Symposium on Research in Computer Security*. Springer, 2020, pp. 480–501.

[94] G. Sun, Y. Cong, J. Dong, Q. Wang, and J. Liu, “Data poisoning attacks on federated machine learning,” *arXiv preprint arXiv:2004.10020*, 2020.

[95] C. Xie, K. Huang, P.-Y. Chen, and B. Li, “Dba: Distributed backdoor attacks against federated learning,” in *International Conference on Learning Representations*, 2019.

[96] F. Sattler, K.-R. Müller, T. Wiegand, and W. Samek, “On the byzantine robustness of clustered federated learning,” in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 8861–8865.

[97] M. Fang, X. Cao, J. Jia, and N. Gong, “Local model poisoning attacks to byzantine-robust federated learning,” in *29th {USENIX} Security Symposium ({USENIX} Security 20)*, 2020, pp. 1605–1622.

[98] A. N. Bhagoji, S. Chakraborty, P. Mittal, and S. Calo, “Analyzing federated learning through an adversarial lens,” in *International Conference on Machine Learning*. PMLR, 2019, pp. 634–643.

[99] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, “How to backdoor federated learning,” in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2020, pp. 2938–2948.

[100] S. Li, Y. Cheng, W. Wang, Y. Liu, and T. Chen, “Learning to detect malicious clients for robust federated learning,” *arXiv preprint arXiv:2002.00211*, 2020.

[101] Z. Wu, Q. Ling, T. Chen, and G. B. Giannakis, “Federated variance-reduced stochastic gradient descent with robustness to byzantine attacks,” *IEEE Transactions on Signal Processing*, vol. 68, pp. 4583–4596, 2020.

[102] A. Mallik, “Man-in-the-middle-attack: Understanding in simple words,” *Cyberspace: Jurnal Pendidikan Teknologi Informasi*, vol. 2, no. 2, pp. 109–134, 2019.

[103] J. Lin, M. Du, and J. Liu, “Free-riders in federated learning: Attacks and defenses,” *arXiv preprint arXiv:1911.12560*, 2019.

[104] X. Xu and L. Lyu, “Towards building a robust and fair federated learning system,” *arXiv preprint arXiv:2011.10464*, 2020.

[105] A. Richardson, A. Filos-Ratsikas, and B. Faltings, “Budget-bounded incentives for federated learning,” in *Federated Learning*. Springer, 2020, pp. 176–188.

[106] W. Sun, N. Xu, L. Wang, H. Zhang, and Y. Zhang, “Dynamic digital twin and federated learning with incentives for air-ground networks,” *IEEE Transactions on Network Science and Engineering*, 2020.

[107] Y. Wang, Z. Su, N. Zhang, and A. Benslimane, “Learning in the air: secure federated learning for uav-assisted crowdsensing,” *IEEE Transactions on Network Science and Engineering*, 2020.

[108] L. Feng, Z. Yang, S. Guo, X. Qiu, W. Li, and P. Yu, “Two-layered blockchain architecture for federated learning over mobile edge network,” *IEEE Network*, 2021.

[109] S. Wang, “Blockfedml: Blockchained federated machine learning systems,” in *2019 International Conference on Intelligent Computing, Automation and Systems (ICICAS)*. IEEE, 2019, pp. 751–756.

[110] J. Weng, J. Weng, J. Zhang, M. Li, Y. Zhang, and W. Luo, “Deepchain: Auditable and privacy-preserving deep learning with blockchain-based incentive,” *IEEE Transactions on Dependable and Secure Computing*, pp. 1–1, 2019.

[111] X. Bao, C. Su, Y. Xiong, W. Huang, and Y. Hu, “Flchain: A blockchain for auditable federated learning with trust and incentive,” in *2019 5th International Conference on Big Data Computing and Communications (BIGCOM)*, 2019, pp. 151–159.

[112] Y. Zhao, J. Zhao, L. Jiang, R. Tan, D. Niyato, Z. Li, L. Lyu, and Y. Liu, “Privacy-preserving blockchain-based federated learning for IoT devices,” *IEEE Internet of Things Journal*, pp. 1–1, 2020.

[113] Y. Li, C. Chen, N. Liu, H. Huang, Z. Zheng, and Q. Yan, “A blockchain-based decentralized federated learning framework with committee consensus,” *IEEE Network*, p. 1–8, 2020. [Online]. Available: http://dx.doi.org/10.1109/MNET.011.2000263

[114] Y. Qu, S. R. Pokhrel, S. Garg, L. Gao, and Y. Xiang, “A blockchained federated learning framework for cognitive computing in industry 4.0 networks,” *IEEE Transactions on Industrial Informatics*, 2020.

[115] Y. Liu, K. Li, Y. Zhang, and W. Qu, “A novel reputation computation model based on subjective logic for mobile ad hoc networks,” in *2009 Third International Conference on Network and System Security*, 2009, pp. 294–301.

[116] M. Barreno, B. Nelson, A. D. Joseph, and J. D. Tygar, "The secuirty of machine learning," *Machine Learning*, vol. 81, pp. 121 – 148, 2010.

[117] C. Fung, C. J. M. Yoon, and I. Beschastnikh, "Mitigating sybils in federated learning poisoning," *arXiv preprint arXiv:1808.04866*, 2020.

[118] M. Shayan, C. Fung, C. J. M. Yoon, and I. Beschastnikh, "Biscotti: A ledger for private and secure peer-to-peer machine learning," *arXiv preprint arXiv:1811.09904*, 2019.

[119] M. Nofer, P. Gomber, O. Hinz, and D. Schiereck, "Blockchain," *Business and Information Systems Engineering: The International Journal of WIRTSCHAFTSINFORMATIK*, vol. 59, no. 3, pp. 183–187, 2017. [Online]. Available: https://EconPapers.repec.org/RePEc:spr:binfse:v:59:y:2017:i:3:d:10.1007_s12599-017-0467-3

[120] S.-J. Hahn and J. Lee, "Privacy-preserving federated bayesian learning of a generative model for imbalanced classification of clinical data," *arXiv preprint arXiv:1910.08489*, 2019.

[121] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-IID data," *arXiv preprint arXiv:1806.00582*, 2018.

[122] T. Luo, S. S. Kanhere, H. Tan, F. Wu, and H. Wu, "Crowdsourcing with tullock contests: A new perspective," in *2015 IEEE Conference on Computer Communications (INFOCOM)*, 2015, pp. 2515–2523.

[123] T. Luo, S. S. Kanhere, J. Huang, S. K. Das, and F. Wu, "Sustainable incentives for mobile crowdsensing: Auctions, lotteries, and trust and reputation systems," *IEEE Communications Magazine*, vol. 55, no. 3, pp. 68–74, 2017.

[124] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *arXiv preprint arXiv:1312.6199*, 2013.

[125] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[126] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, "The limitations of deep learning in adversarial settings," in *2016 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, 2016, pp. 372–387.

[127] A. Qayyum, A. Ijaz, M. Usama, W. Iqbal, J. Qadir, Y. Elkhatib, and A. Al-Fuqaha, "Securing machine learning in the cloud: A systematic review of cloud machine learning security," *Frontiers in big Data*, vol. 3, 2020.

[128] N. Ford, J. Gilmer, N. Carlini, and D. Cubuk, "Adversarial examples are a natural consequence of test error in noise," *arXiv preprint arXiv:1901.10513*, 2019.

[129] N. Papernot, P. McDaniel, A. Sinha, and M. Wellman, "Towards the science of security and privacy in machine learning," *arXiv preprint arXiv:1611.03814*, 2016.