A POTENTIAL GAME PERSPECTIVE IN FEDERATED LEARNING

KANG LIU [1], ZIQI WANG [2,*], AND ENRIQUE ZUAZUA [2,3,4]

ABSTRACT. Federated learning (FL) is an emerging paradigm for training machine learning models across distributed clients. Traditionally, in FL settings, a central server assigns training efforts (or strategies) to clients. However, from a market-oriented perspective, clients may independently choose their training efforts based on rational self-interest. To explore this, we propose a potential game framework where each client's payoff is determined by their individual efforts and the rewards provided by the server. The rewards are influenced by the collective efforts of all clients and can be modulated through a reward factor. Our study begins by establishing the existence of Nash equilibria (NEs), followed by an investigation of uniqueness in homogeneous settings. We demonstrate a significant improvement in clients' training efforts at a critical reward factor, identifying it as the optimal choice for the server. Furthermore, we prove the convergence of the best-response algorithm to compute NEs for our FL game. Finally, we apply the training efforts derived from specific NEs to a real-world FL scenario, validating the effectiveness of the identified optimal reward factor.

1. INTRODUCTION

Federated learning (FL) [24, 42] has shown considerable promise in training large-scale machine learning models across distributed clients without compromising the privacy of their raw data. In FL, clients train local models using their respective local datasets and transmit only the updated model parameters, rather than the raw data, to the central server for aggregation.

To improve learning performance such as model accuracy, it is essential to enlist a sufficient number of clients with diverse data sources [14]. Existing research on FL algorithm design primarily focuses on optimizing learning performance, assuming that enough clients are willing to participate and contribute to the training process [20, 24, 32]. However, in practical scenarios, clients may be hesitant to engage in FL without appropriate compensation due to the following reasons. Firstly, training a local model entails the consumption of the client's local computational resources and data (although not in raw form). Secondly, clients also face the potential risk of data leakage, as FL remains vulnerable to privacy threats such as data reconstruction attacks [29, 43]. Hence, an incentive mechanism is necessary to incentivize clients to produce the desired local updates and allocate rewards by leveraging clients' contributions [36]. In FL, self-interested clients are more incentivized to contribute when offered higher rewards. Meanwhile, the server may be constrained by a limited monetary budget to incentivize clients. To address this multi-agent interaction and decision-making problem, economic and game-theoretic approaches [3, 27] have gained significant attention.

[1] INSTITUT DE MATHÉMATIQUES DE BOURGOGNE, UNIVERSITÉ DE BOURGOGNE, CNRS, 9 AV. ALAIN SAVARY, 21000 DIJON, FRANCE.

[2] CHAIR FOR DYNAMICS, CONTROL, MACHINE LEARNING, AND NUMERICS (ALEXANDER VON HUMBOLDT PROFESSORSHIP), DEPARTMENT OF MATHEMATICS, FRIEDRICH–ALEXANDER-UNIVERSITÄT ERLANGEN–NÜRNBERG, 91058 ERLANGEN, GERMANY.

[3] DEPARTAMENTO DE MATEMÁTICAS, UNIVERSIDAD AUTÓNOMA DE MADRID, 28049 MADRID, SPAIN.

[4] CHAIR OF COMPUTATIONAL MATHEMATICS, FUNDACIÓN DEUSTO. AV. DE LAS UNIVERSIDADES, 24, 48007 BILBAO, BASQUE COUNTRY, SPAIN.

[*] CORRESPONDING AUTHOR.

E-mail addresses: kang.liu@u-bourgogne.fr, ziqi.wang@fau.de, enrique.zuazua@fau.de.

1.1. **Problem formulation and main results.** The task of FL can be formulated as the following minimization problem:

$$(1.1) \qquad \min_{\theta \in \mathbb{R}^n} \sum_{i=1}^m \rho_i F_i(\theta), \ \text{where} \ \rho_i = \frac{D_i}{D}, \ \text{for} \ i \in [m].$$

Here, m denotes the number of clients, $[m] := \{1, \ldots, m\}$ is the set of client indices, $\theta \in \mathbb{R}^n$ represents the trainable parameters, D_i is the size of the dataset for client i, and $D = \sum_{i=1}^m D_i$ represents the total size of all datasets combined. The function F_i denotes the local loss function for client i, typically defined as the empirical risk over its private dataset.

For solving (1.1), the training paradigm of FL involves multiple communication rounds, in which clients perform local training while the server conducts global aggregation. To be concrete, at round $t \in [T]$, each client $i \in [m]$ performs local training in parallel as follows:

$$(1.2) \qquad \theta_i^{t+1} = \text{Update}_i(\theta^t, s_i^t),$$

where Update_i is an abstract update rule that depends on global model parameters θ^t and the client's training effort (or strategy) s_i^t (e.g., the number of local epochs, gradient descent steps, or mini-batch sizes, see Remark 2.1 for more details). After completing the local training, each client sends its updated local model parameters θ_i^{t+1} back to the server for aggregation. For instance, in the most commonly used FedAvg algorithm [24], the server aggregates these parameters using $\theta^{t+1} = \sum_{i=1}^m \rho_i \theta_i^{t+1}$. Finally, the server broadcasts θ^{t+1} to each client for the next round of training.

To obtain better training performance within a fixed number of communication rounds, a commonly adopted approach is to increase clients' training efforts at each round, which has been shown effective in various FL algorithms across different types of neural networks and datasets [14, 20, 24]. Therefore, we make a heuristic assumption in this article that FL training performance is positively correlated with clients' average training effort $\bar{s}$, defined as:

$$(1.3) \qquad \bar{s} = \sum_{i=1}^m \rho_i \bar{s}_i, \ \text{where} \ \bar{s}_i = \frac{1}{T} \sum_{t=1}^T s_i^t.$$

In a typical FL training process, the server assigns the clients' local training efforts, which is analogous to a planned economic system. However, given the inherent limitations and varying capabilities of each client, market-oriented decision-making can be more practical. For example, smaller clients may lack the necessary computational resources to fully implement the training efforts assigned by the server.

Following this market-oriented perspective, in this work, we employ a finite-player game framework to capture the rational training efforts of clients. In this game, each client seeks to maximize its payoff function, which is determined by its local training cost and the reward received from the server. We denote this game as Γ_{FL} and provide its definition below:

Definition 1.1 (FL game Γ_{FL}). The FL game Γ_{FL} contains:
- A set of players (clients) $[m] := \{1, \ldots, m\}$;
- Each client's strategy set $\mathcal{S}_i \subseteq \mathbb{R}_+^T$;
- Each client's payoff function $P_i : \mathcal{S} \to \mathbb{R}$, with $\mathcal{S} = \prod_{i \in [m]} \mathcal{S}_i$, given by

$$(1.4) \qquad P_i(s_i, s_{-i}) = \sum_{t=1}^T r_i(s_i^t, s_{-i}^t) - c_i(s_i^t).$$

In the payoff function (1.4), $c_i(s_i^t)$ is the training or computational cost incurred by client i for employing the effort s_i^t, while $r_i(s_i^t, s_{-i}^t)$ denotes the reward offered by the server to compensate the client's cost. The reward depends on both the client's effort s_i^t and the efforts s_{-i}^t of other clients.

Specifically, the cost $c_i(\cdot)$ and the reward $r_i(\cdot, \cdot)$ considered in this article have the following quadratic forms (see Remarks 2.2 and 2.3 for further explanations):

$$(1.5) \qquad\qquad c_i(s_i^t) = \alpha_i(s_i^t)^2,$$

$$(1.6) \qquad\qquad r_i(s_i^t, s_{-i}^t) = p^t s_i^t, \text{ where } p^t = \lambda \sum_{j=1}^{m} \rho_j s_j^t.$$

Here, p^t denotes the unit price paid by the server based on an aggregative value of all clients' training efforts $\{s_i^t\}_{i \in [m]}$. The constants $\alpha_i > 0$ and $\lambda > 0$ represent the rates at which the marginal costs and payments increase for the clients and the server, respectively. We emphasize that, due to the positivity of λ, the unit price increases with the aggregate of clients' efforts, aligning with our heuristic assumption of a positive correlation between FL training performance and clients' efforts. This contrasts with the classic Cournot competition [8], where the price decreases as aggregate effort or production increases. As a result, the uniqueness of the Nash equilibrium (NE) in our FL game is technical and requires careful analysis (see assumptions of [34, Thm. 1]). An illustration of the cost-intensive mechanism in this game is shown in Figure 2.1.

Although our game model described by (1.4)-(1.6) is inspired by FL, it has broader real-world applications and can extend to various contexts that share this cost-incentive structure. This structure is also common in networks with positive consumption externalities [15], such as technology and social media platforms, where the value of the platform to each user grows as more users join. Consequently, these platforms often design incentives that scale with total engagement, reinforcing collective participation.

In games with the aforementioned structure, the server can adjust the reward factor λ in (1.6) to regulate the NE and the corresponding average training effort $\bar{s}$ defined in (1.3). A natural question arises: How can the server choose a reward factor λ that allocates a reasonable budget to achieve a higher $\bar{s}$, thereby improving training performance? Therefore, in this article, a key challenge we address is to determine an optimal reward factor λ^*. Meanwhile, we investigate typical game-theoretic concerns, such as the existence and uniqueness of the NE, as well as numerical algorithms for computing it.

The main results of this article, addressing the above questions, are summarized as follows:

(1) Existence of NEs: In Theorem 2.4, we investigate the existence of an NE for our FL game, denoted by Γ_{FL}, by demonstrating that Γ_{FL} falls within the framework of a potential game.

(2) Uniqueness of the NE: We examine the uniqueness of the NE in a practical scenario, referred to as the homogeneous FL game, where the players' efforts remain invariant across communication rounds, i.e., $s_i^t = s_i^\tau$ for all $t, \tau \in [T]$. In this setting, we prove in Theorem 3.2 that the NE is unique, except in the case where the parameter λ takes on a critical value λ^*, as defined in (3.3). At this critical point λ^*, the game can have infinite NEs.

(3) Critical reward factor λ^*: We analyze the evolution of the average training effort $\bar{s}$ of NEs with respect to λ in Corollary 3.5. The critical reward factor λ^* acts as a jump point, where $\bar{s}$ experiences a sharp increase, making λ^* the optimal choice for the server's reward factor. Additionally, we introduce two other thresholds, λ_1 and λ_2 (see (3.7)), which represent the activation and saturation points for the training efforts, as illustrated in Figure 1.1.

(4) Algorithmic and numerical validations: We show the convergence of the best-response algorithm for computing NEs in the proposed FL games in Theorem 3.7, with a more general convergence result for general potential games proved in Theorem 5.6. We then apply clients' rational training efforts (i.e., the NE of the game) under different reward factors to a real-world FL training scenario. The results show that while increasing the reward factor enhances training performance, the most significant improvement occurs at the critical jump point λ^*, providing strong numerical evidence for the optimality of λ^* (see Figure 4.3).
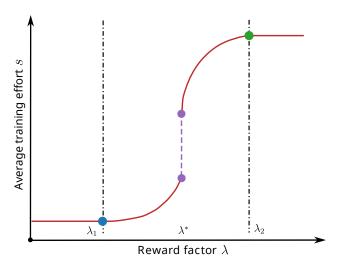
Figure 1.1. Illustration of three critical reward factors: the activation point λ_1, the jump point λ^*, and the saturation point λ_2. See Corollary 3.5 for more details.

It is important to reiterate that our work offers a mathematical perspective on the interplay between clients' training efforts and the server's incentives, paving the way for further exploration of more complex FL systems and networks with positive consumption externalities.

1.2. **Related work.** Current game theory-based incentive mechanisms in FL primarily analyze clients' behaviors by formulating and solving a Stackelberg game [37]. In a Stackelberg game, the players designated as leaders make their moves first, followed by the followers, who act after observing the leaders' decisions. Therefore, it is also termed as a leader-follower two-level game. The solutions to the payoff maximization problems of the leaders and the followers construct the Stackelberg equilibrium.

One of the early explorations of Stackelberg game formulation in FL is in [16], where the server acts as the leader, and the clients are the followers. The server as the leader first sets a monetary reward to the clients. In the lower-level subgame, given the server's reward, each client determines its local training effort to maximize its payoff function. The server is assumed to know the clients' best-response strategy and payoff functions. Then, in the upper-level subgame, the server optimizes a strategy to maximize its payoff based on the best-response of the clients. In other words, the server (leader) in a Stackelberg game needs to solve a bi-level optimization problem, which yields a Stackelberg equilibrium. Similarly, in [19], a two-level Stackelberg game involving multiple mobile edge computing devices as clients and a coordinator as the server is proposed and a unique Stackelberg equilibrium is given as a closed form. Furthermore, [9] extends the FL architecture to a learning market, incorporating service users who are willing to pay for the global model trained by the clients and the server. A three-level hierarchical Stackelberg game model is proposed to simulate the market-oriented interactions among the above three types of participants. The backward induction method [1] is used to derive the analytical Stackelberg equilibrium of the hierarchical Stackelberg game.

Despite the progress mentioned above, the Stackelberg game formulation suffers from several drawbacks. On the one hand, the leader in the upper-level subgame must possess complete information about the followers' payoff functions in the lower-level subgame to derive the Stackelberg equilibrium. This necessitates the disclosure of sensitive and private information by the clients, which contradicts the fundamental privacy principles of FL. On the other hand, existing approaches overlook the constraints of clients' training resources, which can significantly impact their strategies throughout the training process.

To tackle these issues, several works [12, 35, 41] have considered using general non-cooperative games to model FL. However, some mathematical challenges related to NEs, particularly the uniqueness and convergence of computational algorithms, remain insufficiently addressed in these studies. Potential games [25] (see Definition 1.3) offer a promising framework for tackling these challenges. In such games, NEs can be characterized as stationary points of potential functions, simplifying both analysis and computation. A commonly used numerical method to compute NEs in potential games is the best-response algorithm [10, 25, 33], which is based on the best-response mapping introduced by Nash [26]. The potential games, despite their particular structure, have been widely applied to model problems in various fields, such as multi-agent control [22, 5] and resource allocation [38, 28], highlighting their versatility and effectiveness in modeling real-world scenarios. In this work, we explore the possibility of formulating the FL training process within a potential game framework, examining the interplay between clients' local training efforts and the server's incentives.

1.3. **Notations.** In this subsection, we introduce important notations to be used in the following contents. We denote by Γ a game with three components: First, a finite set of players $[m] := \{1, \ldots, m\}$. Second, a pure strategy set $\mathcal{S}_i \subseteq \mathbb{R}^d$ for each player $i \in [m]$, and $\mathcal{S} := \prod_{i \in [m]} \mathcal{S}_i$. Third, a payoff function $P_i : \mathcal{S} \to \mathbb{R}$ for each player $i \in [m]$. We say that a game Γ is finite if its strategy set $\mathcal{S}$ is a finite set. For any $i \in [m]$, we define $\mathcal{S}_{-i} := \prod_{j \neq i} \mathcal{S}_j$. For convenience, for any $s \in \mathcal{S}$, we do not differentiate between $P_i(s)$ and $P_i(s_i, s_{-i})$, where $s_i \in \mathcal{S}_i$ is the i-th coordinate of s and $s_{-i} \in \mathcal{S}_{-i}$ is the tuple of other coordinates.

In a non-cooperative game, each player $i \in [m]$ is considered to be self-interested and only focuses on maximizing its payoff $P_i(s_i, s_{-i})$, which leads to the following concept of the NE.

Definition 1.2 (Nash equilibrium). In game Γ, we call point $s^* \in \mathcal{S}$ an NE if the following inequality holds:
$$P_i\left(s_i^*, s_{-i}^*\right) \geqslant P_i\left(s_i, s_{-i}^*\right), \quad \forall s_i \in \mathcal{S}_i, \forall i \in [m].$$

A special type of non-cooperative games is known as potential games [25], which can be defined as follows:

Definition 1.3 (Potential game). Let $w = (w_i)_{i \in [m]}$ be a strictly positive vector. The game Γ is said to be a (w-, or weighted) potential game if there exists a function $P : \mathcal{S} \to \mathbb{R}$ such that
$$P_i(s_i, s_{-i}) - P_i(x_i, s_{-i}) = w_i \left(P(s_i, s_{-i}) - P(x_i, s_{-i})\right), \quad \forall s_i, x_i \in \mathcal{S}_i, \forall i \in [m].$$

1.4. **Outline of the paper.** The remainder of the paper is organized as follows. In Section 2, we model the FL training process as a potential game and demonstrate the existence of the NE. Next, in Section 3, we focus on the uniqueness of the NE in a homogeneous scenario and study the convergence of the best-response algorithm. The simulation results and analysis are presented in Section 4, followed by detailed technical proofs and theorems in Section 5. Finally, we summarize our conclusions and perspectives in Section 6.

2. A POTENTIAL GAME FRAMEWORK FOR FEDERATED LEARNING

In this section, we first provide a detailed explanation of the cost-incentive mechanism described in (1.5)-(1.6) from the introduction. Next, we introduce the corresponding game formulation as described in Definition 1.1. The existence of NEs is then established based on the potential structure of the game, as shown in Theorem 2.4.

2.1. **Training-cost-incentive workflow in federated learning.** As a distributed system designed to solve the optimization problem (1.1), the conventional FL training process does not provide clients with the flexibility to choose their training efforts. To address this, we introduce a cost-incentive mechanism in (1.5)-(1.6). Let us describe the FL training process, incorporating the

cost-incentive mechanism, in the following summarized workflow. A graphical illustration is also provided in Figure 2.1.

At each communication round $t \in [T]$, the training-cost-incentive workflow of FL is outlined as follows:

Clients side: In parallel, each client $i \in [m]$ selects a training effort s_i^t and performs local training (see Remark 2.1).

- The client executes local training using the global model parameters θ^t and its effort s_i^t:

$$\theta_i^{t+1} = \text{Update}_i(\theta^t, s_i^t);$$

- The client evaluates the local cost based on its prescribed cost coefficient α_i and effort s_i^t:

$$c_i(s_i^t) = \alpha_i(s_i^t)^2.$$

Then, the client sends the updated local model parameters θ_i^{t+1} along with his training effort s_i^t to the server for aggregation.

Server side: The server aggregates clients' local model parameters and training efforts using weights $\rho_i = D_i/D$ defined in (1.1), which represent the proportion of client i's data size D_i relative to the total data size D across all clients.

- The server updates the global model parameters:

$$\theta^{t+1} = \sum_{i=1}^{m} \rho_i \theta_i^{t+1};$$

- The server determines the unit rewarding price using a chosen reward factor λ:

$$p^t = \lambda \sum_{i=1}^{m} \rho_i s_i^t.$$

Then, the server pays each client $i \in [m]$ a reward of $p^t s_i^t$ and simultaneously broadcasts θ^{t+1} to each client for the next round of training.
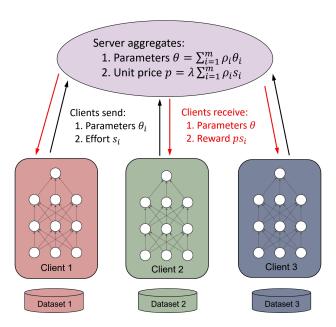

Figure 2.1. Training-cost-incentive workflow of FL, with the round index t omitted.

In the following remarks, we discuss the role of the training effort s_i^t and the rationale behind the quadratic cost-incentive mechanism (1.5)-(1.6).

Remark 2.1 (Local training effort s_i^t in (1.2)). The client's local update (1.2) typically consists of multiple (stochastic) gradient descent steps. For instance, let the training effort s_i^t correspond to the number of gradient descent steps, the client initializes $\theta_i^{t,0} = \theta^t$ and then performs gradient descent for s_i^t times using

$$\theta_i^{t,e} = \theta_i^{t,e-1} - \eta_i \nabla F_i(\theta_i^{t,e-1}), \quad \text{for } e = 1, \ldots, s_i^t,$$

where η_i denotes the learning rate (also known as the step size). After completing the local training, the client sets $\theta_i^{t+1} = \theta_i^{t,s_i^t}$ and sends it back to the server for aggregation.

In practice, training efforts are often measured in epochs rather than steps, with one epoch representing a full pass through a client's training dataset. In this work, we assume that all clients have a consensus on the definition of the local training effort.

Remark 2.2 (Quadratic structure of c_i in (1.5)). The use of a quadratic cost function is common in economic contexts, as it effectively models situations where costs increase disproportionately with greater levels of effort or production. For example, the cost of overproducing a product or exerting excessive effort in competition may rise quadratically, reflecting diminishing returns or increasing risks [17]. Following this, we propose $c_i(s_i^t) = \alpha_i(s_i^t)^2$ as a quadratic function of the training effort s_i^t, where the positive coefficient α_i represents the rate at which the client's marginal cost increases.

Remark 2.3 (Quadratic structure of r_i in (1.6)). For the incentive in (1.6), it is natural for the reward to be defined as $r_i = p^t s_i^t$, representing the product of the unit price p^t and the client's effort s_i^t. Here, $p^t = \lambda \sum_{i=1}^m \rho_i s_i^t$ is positively proportional to the weighted sum of all clients' training efforts, with weights ρ_i defined in (1.1). This monotonic relationship aligns with the server's goal of encouraging clients to increase their training efforts, thereby achieving improved training performance. Note that the server can influence the unit price p^t by adjusting λ. A key challenge addressed in this work is to determine an appropriate reward factor λ that balances budget constraints and performance goals.

2.2. **Existence of Nash equilibrium of $\Gamma_{\mathbf{FL}}$.** Considering the cost-incentive mechanism provided by (1.5)-(1.6), we arrive at formulating an aggregative game (see [13, 31]) for the FL training process, which is referred to as Γ_{FL} (see Definition 1.1) in this paper.

Theorem 2.4 (Existence). *The game Γ_{FL} is a (weighted) potential game with the potential given by*

$$(2.1) \qquad P_{\mathrm{FL}}(s) = \sum_{i=1}^m \sum_{t=1}^T \left(\left(\frac{\lambda \rho_i^2}{2} - \alpha_i \rho_i \right) (s_i^t)^2 \right) + \sum_{t=1}^T \frac{\lambda}{2} \left(\sum_{i=1}^m \rho_i s_i^t \right)^2, \quad s \in \mathcal{S},$$

and $w_i = 1/\rho_i$ for $i \in [m]$. If $\mathcal{S}$ is compact, then Γ_{FL} possesses at least one NE.

Proof. It is straightforward to verify that P_{FL} given in (2.1) is a w-potential of Γ_{FL} in the sense of Definition 1.3, with $w_i = 1/\rho_i$ for all $i \in [m]$. If $\mathcal{S}$ is compact, then P_{FL} admits a maximizer in $\mathcal{S}$, which is an NE of Γ_{FL}, see [25, Lem. 2.1]. $\square$

The compactness of $\mathcal{S}$ is a crucial condition for the existence of NEs. This assumption is highly realistic in FL, and we illustrate this with two common examples where $\mathcal{S}$ is compact.

Example 2.5 (Total budget case). FL is widely used to train models across distributed clients, often personal electronic devices [2]. For example, one of the earliest implementations by Google involved training next-word prediction models using computational resources from clients' smartphones [11].

However, smartphones face practical constraints, such as limited battery life. To account for these limitations, we define the strategy set $\mathcal{S}_i$ for each client as follows:

$$(2.2) \qquad \mathcal{S}_i = \left\{ s_i \in \mathbb{R}_+^T \,\middle|\, b_i \leq \sum_{t \in [T]} s_i^t \leq B_i \right\},$$

where b_i and B_i represent the minimum and maximum total training efforts during the entire training process for client $i \in [m]$. These limits are determined by the minimum training requirements and the client's budget constraints.

Example 2.6 (Homogeneous case). In FL, it is common for clients to set their training effort at the beginning and maintain it across multiple training rounds [24]. This approach offers practical advantages, such as reducing fluctuations in local computational power and ensuring stability throughout the training process. In this context, we consider the strategy set $\mathcal{S}_i$ as follows:

$$(2.3) \qquad \mathcal{S}_i = \left\{ s_i \in \mathbb{R}_+^T \,|\, q_i \leq s_i^t = s_i^\tau \leq Q_i, \, \forall t, \tau \in [T] \right\},$$

where q_i and Q_i are minimum and maximum training efforts at each round for client $i \in [m]$.

In addition to the existence result presented in Theorem 2.4, establishing the uniqueness of the NE of Γ_{FL} is often more technically involved. A straightforward scenario for uniqueness can be derived from the strict concavity of the potential function $P_{\mathrm{FL}}(s)$, but this requires the coefficients α_i to be sufficiently large, e.g., $\alpha_i > \lambda(1 + \rho_i^{-1})/2$ for all i, which is not practical in most cases. In more general settings where α_i and λ do not satisfy such constraints, the uniqueness is easily lost, especially in heterogeneous scenarios like Example 2.5, see Remark 3.10 for more details. In contrast, for the homogeneous case (Example 2.6), we establish in the following section the uniqueness of the NE for all values of λ, except at the jump point given by (3.3). This result is derived through a detailed analysis of the fixed-point system characterizing the NE, as presented in Section 5.

3. UNIQUENESS RESULTS AND THE BEST-RESPONSE ALGORITHM

In this section, we explore the uniqueness of the NE in the homogeneous case of Γ_{FL} (Example 2.6). We begin by reformulating this scenario in Definition 3.1. Then, we present results on the uniqueness of the NE in this setting. Following this, we introduce the best-response algorithm for computing the NE and prove its convergence. At the end of this section, we offer discussions on extensions to non-uniform datasets and heterogeneous scenarios.

3.1. **Homogeneous game formulation.** To simplify the notation in the statements of theorems and proofs, we assume that all clients' datasets are of uniform size, i.e., $D_i = D_j$ for any i and j. For the non-uniform case, we refer to an analogous analysis in Remark 3.9. The game Γ_{FL} in Definition 1.1, with the strategy sets given by (2.3), can be reformulated equivalently as follows (denoted by Γ_{FL}^h):

Definition 3.1 (Homogeneous FL game Γ_{FL}^h). The homogeneous FL game Γ_{FL}^h contains:

- A set of players (clients) $[m] := \{1, \ldots, m\}$;
- Each client's strategy set $\mathcal{S}_i^h = [q_i, Q_i]$;
- Each client's payoff function $P_i^h : \mathcal{S}^h \to \mathbb{R}$, with $\mathcal{S}^h = \prod_{i \in [m]} \mathcal{S}_i^h$, given by

$$(3.1) \qquad P_i^h(s_i, s_{-i}) = \frac{\lambda}{m} \sum_{j=1}^m s_j s_i - \alpha_i (s_i)^2.$$

As a special case of the game Γ_{FL}, it follows from Theorem 2.4 that the homogeneous game Γ_{FL}^h is a potential game and admits NEs. Moreover, the homogeneous game allows us to study the uniqueness of the NE, which is more challenging in a heterogeneous setting, as discussed at the end of Section 2 and Remark 3.10.

3.2. Uniqueness of the NE of the homogeneous game.
Let us introduce crucial constants for the uniqueness result in this subsection:

- Concavity threshold:

$$\text{(3.2)} \qquad \bar{\lambda} = m\alpha_{\min}, \text{ where } \alpha_{\min} := \min_{i \in [m]}\{\alpha_i\};$$

- Critical point λ^* (jump point), which is the unique solution of the following equation in the interval $(0, \bar{\lambda})$:

$$\text{(3.3)} \qquad \sum_{i=1}^{m} \frac{\lambda^*}{2m\alpha_i - \lambda^*} = 1;$$

- Two constants c_1 and c_2 to determine if a jump occurs at the critical point λ^*:

$$\text{(3.4)} \qquad c_1 := \max_{i \in [m]}\left\{\frac{2\alpha_i q_i}{\lambda^*} - \frac{q_i}{m}\right\}, \quad c_2 := \min_{i \in [m]}\left\{\frac{2\alpha_i Q_i}{\lambda^*} - \frac{Q_i}{m}\right\}.$$

For the jump point role of λ^*, we refer to Figure 1.1.

Theorem 3.2 (Uniqueness). *The following statements hold:*

 (1) *If $\lambda > 0$ and $\lambda \neq \lambda^*$, then the game Γ_{FL}^h has a unique NE.*
 (2) *If $\lambda = \lambda^*$, then the following holds:*
  (a) *If $c_1 \geqslant c_2$, then the game Γ_{FL}^h has a unique NE;*

  (b) *If $c_1 < c_2$, then the game Γ_{FL}^h has infinite NEs. A point $s^* \in \mathcal{S}^h$ is an NE if and only if there exists $c \in [c_1, c_2]$ such that*

$$\text{(3.5)} \qquad s_i^* = \frac{\lambda^* c}{2\alpha_i - \lambda^*/m}, \quad \forall i \in [m].$$

Proof. The proof is presented in Section 5. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ $\square$

Remark 3.3. We refer to $\bar{\lambda}$ as the concavity threshold, because when $\lambda \in (0, \bar{\lambda})$, each player's payoff function $P_i^h(s_i, s_{-i})$ is concave with respect to their strategy s_i. In this case, our game Γ_{FL}^h falls within the framework of concave games [30], which is commonly used in the literature for studying uniqueness. In our setting, thanks to the quadratic form of $P_i^h(s_i, s_{-i})$, we can handle the non-concave case as well. As stated in Theorem 3.2 (1), the range of λ can extend beyond $\bar{\lambda}$. We introduce $\bar{\lambda}$ in (3.2) specifically for defining the jump point λ^*.

Remark 3.4. In this remark, we present a special symmetric case such that $\bar{\lambda}$, λ^*, c_1, and c_2 have explicit expressions. Assume that $\alpha_i = \alpha_j = \alpha$ for all $i, j \in [m]$, then we have

$$\text{(3.6)} \qquad \bar{\lambda} = m\alpha, \quad \lambda^* = \frac{2m}{m+1}\alpha, \quad c_1 = \max_{i \in [m]}\{q_i\}, \quad c_2 = \min_{i \in [m]}\{Q_i\}.$$

In this case, it is common for $c_1 < c_2$, since the minimum effort q_i is usually set to a uniformly small value. For example, let the client's training effort be the number of local epochs, the minimum effort q_i can be set uniformly to one while the maximum effort $Q_i > 1$. Moreover, this symmetric case provides an important perspective for extending our FL game to the mean-field context [6, Section 2], i.e., an infinite number of players with symmetric costs.

In the following corollary, we present a more precise description of the unique NE of game Γ_{FL}^h when $\lambda \neq \lambda^*$. Before that, we define two additional critical thresholds, namely the activation point λ_1 and the saturation point λ_2 (see Figure 1.1), as follows:

$$(3.7) \qquad \lambda_1 := 2 \min_{i \in [m]} \left\{ \frac{\alpha_i q_i}{\bar{q} + \frac{q_i}{m}} \right\}, \quad \lambda_2 := 2 \max_{i \in [m]} \left\{ \frac{\alpha_i Q_i}{\bar{Q} + \frac{Q_i}{m}} \right\},$$

where q_i and Q_i are the minimum and maximum training efforts of each client $i \in [m]$ (see Definition 3.1), while $\bar{q} = \sum_{i=1}^m q_i/m$ and $\bar{Q} = \sum_{i=1}^m Q_i/m$ are the average minimum and maximum efforts across all clients.

It can be deduced from their definitions that

$$0 < \lambda_1 \le \lambda^* \le \lambda_2.$$

Corollary 3.5 (Precise description of the unique NE). *The following statements hold:*

(1) If $\lambda \in (0, \lambda_1)$, then the unique NE of the game Γ_{FL}^h is $s^* = (q_i)_{i \in [m]}$.

(2) If $\lambda \in (\lambda_1, \lambda^*)$, then the unique NE of the game Γ_{FL}^h, denoted by s^*, satisfies

$$\bar{s}^* := \frac{1}{m} \sum_{i \in [m]} s_i^* < c_1.$$

(3) If $\lambda \in (\lambda^*, \lambda_2)$, the unique NE of the game Γ_{FL}^h, denoted by s^*, satisfies

$$\bar{s}^* := \frac{1}{m} \sum_{i \in [m]} s_i^* > c_2.$$

(4) If $\lambda \in (\lambda_2, +\infty)$, the unique NE of the game Γ_{FL}^h is $s^* = (Q_i)_{i \in [m]}$.

Proof. The proof is presented in Section 5. □

Remark 3.6. As discussed in Remark 3.4, it is important to note that $c_1 < c_2$ is a common scenario in FL training. From Corollary 3.5 (2)-(3), it follows that increasing the reward factor beyond λ^* results in a significant jump in the average effort $\bar{s}^*$ (see Figure 1.1). As demonstrated later through numerical tests in Section 4, this jump point λ^* represents the optimal reward factor for the server to select (see Table 4.2 and Figure 4.3).

3.3. **Computation of NEs via the best-response algorithm.** The best-response algorithm [10, 25, 33] is a widely employed and easily implemented numerical method to find an NE for potential games. Here, we present the best-response algorithm adapted to our framework, as outlined in Algorithm 1.

A classical convergence result of Algorithm 1 pertains to finite potential games [25, Cor. 2.2, Lem. 2.3]. We extend this result to continuous potential games under certain assumptions, as detailed in Assumption 5.4. These results offer convergence guarantees for Algorithm 1 when applied to our FL games.

Recall the definitions of $\bar{\lambda}$ and λ^* from (3.2) and (3.3). We have the following theorem on the convergence of Algorithm 1 applied to the game Γ_{FL}^h.

Theorem 3.7 (Convergence of Algorithm 1). *Assume that $\lambda \in (0, \bar{\lambda})$ and $\lambda \neq \lambda^*$. For $k \ge 1$, let s^k be the result of Algorithm 1 at k-th iteration. Then, the sequence $\{s^k\}_{k \ge 1}$ converges to the unique NE of Γ_{FL}^h.*

Proof. The game Γ_{FL}^h satisfies Assumption 5.4. By applying the general convergence result of the best-response algorithm from Theorem 5.6 (2), we obtain that any limit point of the sequence $\{s^k\}_{k \ge 1}$ is an NE of Γ_{FL}^h. By Theorem 3.2, if $\lambda \neq \lambda^*$, this NE is unique, which implies that the limit point of $\{s^k\}_{k \ge 1}$ is unique. Combining this with the pre-compactness of $\{s^k\}_{k \ge 1}$, the conclusion follows. □

Algorithm 1 Best-response algorithm

1: **Initialization:** $s^0 \in \mathcal{S}^h$;
2: **for** $k = 1, \ldots$ **do**
3:     **for** $i = 1, \ldots, m$ **do**
4:         **if** $s_i^{k-1} \notin \arg\max_{s_i \in \mathcal{S}_i^h} P_i^h(s_1^k, \ldots, s_{i-1}^k, s_i, s_{i+1}^{k-1}, \ldots, s_m^{k-1})$ **then**
5:             $s_i^k \in \arg\max_{s_i \in \mathcal{S}_i^h} P_i^h(s_1^k, \ldots, s_{i-1}^k, s_i, s_{i+1}^{k-1}, \ldots, s_m^{k-1})$;
6:         **else**
7:             $s_i^k = s_i^{k-1}$;
8:         **end if**
9:     **end for**
10:     **if** $s^k = s^{k-1}$ **then**
11:         **return** s^k.
12:     **end if**
13: **end for**

Building on Theorem 5.6, the following remark establishes the convergence result of Algorithm 1 for the specific case $\lambda = \lambda^*$ and the convergence rate, serving as a complement to Theorem 3.7.

Remark 3.8. Recall the definitions of c_1 and c_2 from (3.4). If $c_1 \geqslant c_2$, the convergence result in Theorem 3.7 holds for $\lambda = \lambda^*$ (see Theorem 3.2 (2a)). Additionally, if $c_1 < c_2$, at the point $\lambda = \lambda^*$, any cluster point of the sequence $\{s^k\}_{k \geqslant 1}$ is an NE of Γ_{FL}^h. In all these cases, for any $K \geqslant 1$, there exists $k \in \{1, \ldots, K\}$ such that s^k is an $\mathcal{O}(1/K)$-NE of Γ_{FL}^h. Here, we refer to Definition 5.5 for the definition of an ϵ-NE.

3.4. **Discussion on non-uniform data sizes and heterogeneous games.** In the following two remarks, we discuss two complementary cases regarding the homogeneous game described in Definition 3.1.

Remark 3.9 (Non-uniform data sizes). In Definition 3.1, we assume that all clients have equal data sizes to simplify the constants and the analysis in our main results (Theorem 3.2). However, in practice, clients typically possess different data sizes, as introduced in (1.1). Consequently, instead of the payoff functions in (3.1), the associated homogeneous game has the following payoff functions:

$$P_i^h(s_i, s_{-i}) = \lambda \sum_{j=1}^m \rho_j s_j s_i - \alpha_i(s_i)^2, \quad \forall i \in [m].$$

where $\rho_j = D_j/D$ represents the weight of client j's dataset size D_j relative to the total dataset size D across all clients.

In this non-uniform case, the existence of NE is guaranteed by Theorem 2.4. Regarding uniqueness, by following the proof in Section 5, we can establish the same results as in Theorem 3.2 and Corollary 3.5, with the constants introduced in (3.2)-(3.4) and (3.7) replaced by the following:

$$\bar{\lambda} = \min_{i \in [m]} \{\alpha_i/\rho_i\};$$

$$\lambda^* \in (0, \bar{\lambda}), \quad \text{s.t.} \quad \sum_{i=1}^m \frac{\lambda^*}{2\alpha_i/\rho_i - \lambda^*} = 1;$$

$$c_1 = \max_{i \in [m]} \left\{ \frac{2\alpha_i q_i}{\lambda^*} - \rho_i q_i \right\}, \quad c_2 = \min_{i \in [m]} \left\{ \frac{2\alpha_i Q_i}{\lambda^*} - \rho_i Q_i \right\};$$

$$\lambda_1 = 2 \min_{i \in [m]} \left\{ \frac{\alpha_i q_i}{\bar{q} + \rho_i q_i} \right\}, \quad \lambda_2 = 2 \max_{i \in [m]} \left\{ \frac{\alpha_i Q_i}{\bar{Q} + \rho_i Q_i} \right\},$$

where $\bar{q} = \sum_{i \in [m]} \rho_i q_i$ and $\bar{Q} = \sum_{i \in [m]} \rho_i Q_i$. Note that if $\rho_i = 1/m$ for all $i \in [m]$ (i.e., the uniform case), then we recover the constants defined in Section 3.2. In addition, we obtain the same results as in Theorem 3.7 and Remark 3.8 for the convergence of Algorithm 1 in this non-uniform case, since our general convergence result, Theorem 5.6, is established for weighted-potential games.

Remark 3.10 (Heterogeneous games). A key assumption for the uniqueness results presented in this section is the homogeneity of training efforts with respect to the communication rounds, i.e., s_i^t is independent of t. However, if we consider the heterogeneous case (the original game Γ_{FL} with strategy sets (2.2)), the uniqueness can easily be lost. For instance, suppose that there exists a heterogeneous NE s^*, meaning s^* is an NE, and there exists rounds $t_1, t_2 \in [T]$ such that $(s^*)^{t_1} \neq (s^*)^{t_2}$. Then, by the potential function (2.1), we can verify that $\tilde{s}^*$ (which differs from s^*) is also an NE of the game Γ_{FL}, where $\tilde{s}^*$ is obtained by swapping the training efforts of s^* at rounds t_1 and t_2.

Although the uniqueness result does not hold, Algorithm 1 still converges in the heterogeneous case, similar to Remark 3.8. Let s^k denote the outcome of iteration k in Algorithm 1 adapted to Γ_{FL}. Then, any cluster point of s^k is an NE of Γ_{FL}.

4. Numerical experiments

In this section, we conduct numerical simulations of our FL games. We begin by introducing the experimental settings. Then, we compute NEs of the homogeneous game Γ_{FL}^h under different reward factors λ by Algorithm 1. From these simulations, we observe and validate the evolution of NEs with respect to λ, aligning with Corollary 3.5. Finally, we apply clients' rational training efforts (i.e., the NE of the game) associated with thresholds of λ (i.e., $\lambda_1, \lambda^*, \lambda_2$) to a real-world FL scenario, comparing the resulting model performance to validate the effectiveness of our proposed reward factor. Codes used to reproduce the examples can be found at: https://github.com/DCN-FAU-AvH/FL-Potential-Game.

4.1. **Setups.** In this subsection, we elaborate on the experimental settings and also summarize them in Table 4.1. We consider a homogeneous game scenario with $m = 20$ clients and $T = 50$ communication rounds. The server selects a reward factor $\lambda \in [0, 5]$, while each client has a local computational cost factor $\alpha_i \in [1, 2]$. Each client's training effort s_i is defined as the number of local training epochs, with the specific ranges $[q_i, Q_i]$ indicated in the corresponding tests.

Parameters	Values
Dataset	MNIST
Neural network	CNN
Activation function	ReLu
FL algorithm	FedAvg
Number of clients	20
Number of rounds	50

Table 4.1. Experimental scenario and parameter settings.

The FL task we consider is the image classification problem with the MNIST dataset [18]. The MNIST dataset consists of grayscale images of handwritten digits from 0 to 9 (10 labels) with a size of 28×28 pixels. We select $D = 10,000$ training samples, distributing them evenly among 20 clients in a non-independent and identically distributed (non-IID) manner, meaning each client has $D_i = 500$ data points with at most two labels. We also use 10,000 test samples to evaluate the prediction accuracy of the trained model. For the FL algorithm, we use FedAvg [24], assuming that all clients are active in each round, performing s_i epochs of local training with a batch size of 250 and a learning rate of 0.01. The neural network used by each client is a convolutional neural

network (CNN) with two convolutional layers, comprising 32 and 64 channels, respectively, as the same to that in [24]. Each client employs the cross-entropy loss function and the ReLU activation function.

4.2. **The evolution of NEs with respect to different reward factors.** In this subsection, we validate our theoretical results in Section 3.2 by examining the relationship between the clients' average training effort $\bar{s}$ and the server's reward factor λ. We consider two scenarios in which each client has a different range of training efforts. In Scenario 1, the minimum training effort for each client is fixed at $q_i = 1$, while in Scenario 2, q_i is uniformly sampled from the range $[1, 20]$. The maximum training effort is set to $Q_i \in [20, 30]$ in both scenarios, with other parameters remaining the same as indicated in Section 4.1. We compute the NE using Algorithm 1 and present the results in Figure 4.1 and Figure 4.2.

We emphasize that Scenario 1 is more prevalent in practical FL applications than Scenario 2, since the training effort corresponds to the number of local epochs, with a natural lower bound of one and a typical upper bound greater than one. In our simulation for Scenario 1, the constant $c_1 = 1.37$ is less than $c_2 = 14.81$, which aligns with the conditions in Theorem 3.2 (2b). In contrast, in the experiment of Scenario 2, the constant $c_1 = 19.62$ exceeds $c_2 = 14.81$, satisfying the condition in Theorem 3.2 (2a). This comparison highlights distinct behaviors of NEs under differing parameter settings.

As shown in Figure 4.1, in both scenarios, the average effort $\bar{s}$ increases with the reward factor λ and exhibits similar behavior in the initial and final stages. Before the activation point λ_1, $\bar{s}$ remains at its minimum value $\bar{q}$ as shown in Figures 4.2a and 4.2d. After the saturation point λ_2, $\bar{s}$ saturates at $\bar{Q}$ as shown in Figures 4.2c and 4.2f. These trends are well-captured by our theoretical analysis (see Corollary 3.5), indicating that the server should select a reward factor between λ_1 and λ_2 to avoid insufficient or excessive training.

However, the evolution of $\bar{s}$ between λ_1 and λ_2 differs notably between the two scenarios, particularly around the critical jump point λ^*. These differences are outlined as follows:

(1) In Scenario 1, where $c_1 < c_2$, $\bar{s}$ exhibits a significant jump from c_1 to c_2 at λ^*. According to Theorem 3.2 (2b), every point between c_1 and c_2 corresponds to an NE. This implies that if the server sets the reward factor to λ^*, the NE of the game Γ_{FL}^h could occur at any point between c_1 and c_2. Therefore, the optimal strategy for the server is to choose a reward factor slightly beyond λ^*, a finding that is also validated by simulations in the next subsection using real-world FL training examples.

(2) In Scenario 2, where $c_1 > c_2$, the jump phenomenon observed in Scenario 1 does not occur. We numerically observe the uniqueness of the NE at λ^*, aligning with Theorem 3.2 (2a). Nevertheless, as shown in Figure 4.1b and Figure 4.2e, $\bar{s}$ still increases more sharply around λ^* compared to other stages, underscoring its importance as the proper choice for the server's reward factor.

Overall, these results highlight the sensitivity of clients' training efforts to the server's reward factor across different scenarios. We identify the optimal reward factor for the server as a value slightly larger than the jump point λ^*, where a sharp increase in the average training effort is observed. The effectiveness of this choice is further validated by real FL training tests in the next subsection.

4.3. **The training performance of FL across various reward factors.** In this subsection, we evaluate the FL training performance under different reward factors to further validate our proposals for the server's selection of the reward factor λ. Specifically, we select four representative reward factors around the three critical thresholds identified in Scenario 1 (refer to Figure 4.1a), as detailed in Table 4.2. The results of FL training in four cases are presented in Figure 4.3.

These cases examine how varying the server's reward factor λ influences clients' training efforts and, more importantly, the resulting FL training performance. In the tests, we apply clients'
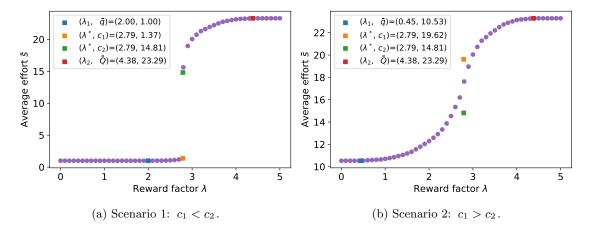
(a) Scenario 1: $c_1 < c_2$.

(b) Scenario 2: $c_1 > c_2$.

Figure 4.1. Results of the clients' average training effort $\bar{s}$ with respect to the server's reward factor λ. Scenario 1 (Left): $q_i \in \{1\}$, $Q_i \in [20, 30]$, $c_1 < c_2$. Scenario 2 (Right): $q_i \in [1, 20]$, $Q_i \in [20, 30]$, $c_1 > c_2$. The values of the activation point λ_1, jump point λ^*, and saturation point λ_2 are indicated in the legends.
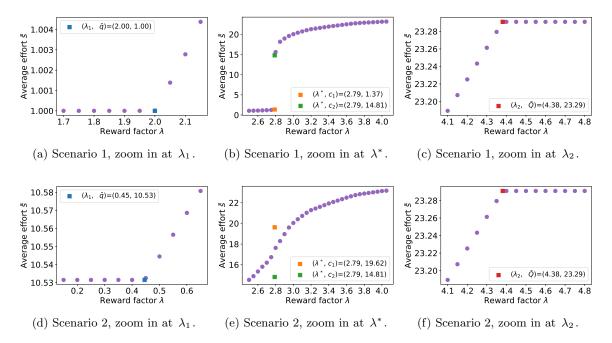

(a) Scenario 1, zoom in at λ_1.

(b) Scenario 1, zoom in at λ^*.

(c) Scenario 1, zoom in at λ_2.

(d) Scenario 2, zoom in at λ_1.

(e) Scenario 2, zoom in at λ^*.

(f) Scenario 2, zoom in at λ_2.

Figure 4.2. Zoom-in of Figure 4.1 around the three critical thresholds: activation point λ_1, jump point λ^*, and saturation point λ_2. Scenario 1 (Top) corresponds to Figure 4.1a, and Scenario 2 (Bottom) corresponds to Figure 4.1b.

rational efforts $(s_i)_{i \in [m]}$ across these four cases to FL training, with detailed training settings provided in Section 4.1. In all tests, each client performs $\lceil s_i \rceil$ epochs of local training, where $\lceil \cdot \rceil$ denotes the ceiling function (i.e., $\lceil x \rceil$ is the smallest integer no less than x). We present in Figure 4.3 the results of FL training, depicted through accuracy and loss curves, across the four different cases. The observations are outlined as follows:

(1) In Case 1 (selected before the activation point λ_1), the model shows the worst performance, with both accuracy and loss curves reflecting suboptimal training due to insufficient clients' training efforts.

	Case 1	Case 2	Case 3	Case 4
λ	1.95	2.75	2.80	4.40
$\bar{s}$	1	1.3	15.6	23.3

Table 4.2. Reward factor λ and average training effort $\bar{s}$ in four test cases shown in Figure 4.3. Case 1 is selected before the activation point ($\lambda_1 = 2$). Cases 2 and 3 are selected before and after the jump point ($\lambda^* = 2.79$), respectively. Case 4 is selected after the saturation point ($\lambda_2 = 4.38$).

(a) Test accuracy.

(b) Training loss.
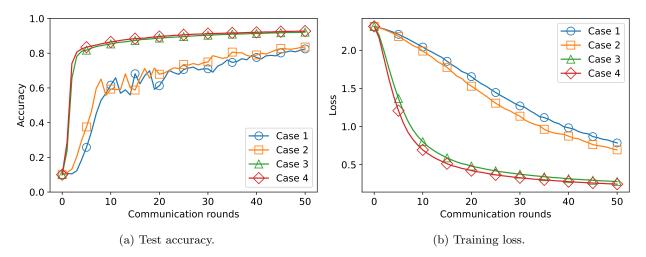
Figure 4.3. FL training performance in four cases with different server reward factors (see Table 4.2). Case 1 is selected before the activation point, Cases 2 and 3 are selected before and after the jump point, respectively, and Case 4 is selected after the saturation point.

(2) In Case 2 (selected before the jump point λ^*), there is a slight improvement in performance compared to Case 1, but the gains are very limited, indicating that the marginal increase in training effort is not enough to significantly enhance model accuracy.

(3) Notably, in Case 3 (selected after the jump point λ^*), there is a significant improvement in training performance, marked by a sharp increase in accuracy and a corresponding decrease in loss.

(4) In Case 4 (selected after the saturation point λ_2), further increases in training effort lead to only marginal improvements in accuracy and a slight reduction in loss. This indicates that beyond the jump point, as clients further increase their training efforts, the improvements in model performance gradually plateau, reflecting diminishing returns.

Overall, these results suggest that while increasing the reward factor λ can improve training performance, the most significant gains are achieved around the jump point λ^*. Beyond this point, additional incentives from the server yield diminishing returns. Therefore, an optimal strategy for the server would be to select a reward factor slightly above the jump point λ^* to maximize model performance while minimizing unnecessary incentive expenditure.

5. Technical proofs and theorems

5.1. **Proofs of Theorem 3.2 and Corollary 3.5.** Recall parameters λ, m, α_i, q_i, and Q_i of the game Γ_{FL}^h from Definition 3.1 and Example 2.6. Recall the formula for the payoff function P_i^h from (3.1). In the following lemma, we derive the fixed point conditions satisfied by the NEs of the game Γ_{FL}^h based on the quadratic formula of P_i^h.

Lemma 5.1. *A point* $s^* \in \prod_{i \in [m]} [q_i, Q_i]$ *is an NE of the game* Γ^h_{FL} *if and only if the following holds:*

$$
\begin{array}{ll}
(5.1a) \\
(5.1b) \qquad s_i^* = \left\{
\begin{array}{ll}
q_i & , \quad \text{if } q_i \geqslant \dfrac{\lambda\left(\bar{s}^* - s_i^*/m\right)}{2\left(\alpha_i - \lambda/m\right)} \text{ and } \alpha_i > \dfrac{\lambda}{m}, \\[4mm]
\dfrac{\lambda \bar{s}^*}{2\alpha_i - \lambda/m}, & \text{if } q_i < \dfrac{\lambda\left(\bar{s}^* - s_i^*/m\right)}{2\left(\alpha_i - \lambda/m\right)} < Q_i, \\[4mm]
Q_i & , \quad \text{if } Q_i \leqslant \dfrac{\lambda\left(\bar{s}^* - s_i^*/m\right)}{2\left(\alpha_i - \lambda/m\right)} \text{ or } \alpha_i \leqslant \dfrac{\lambda}{m},
\end{array}
\right.
\\
(5.1c)
\end{array}
$$

where $\bar{s}^* = \sum_{i \in [m]} s_i^*/m$.

We present equivalent formulations of (5.1a)-(5.1c) in Lemma 5.2, which is utilized in the proof of Theorem 3.2. Before proceeding, let us introduce the following m auxiliary mappings in $\mathbb{R}_+$ for $i \in [m]$:

$$
\begin{array}{ll}
(5.2a) \\
(5.2b) \qquad \beta_i(x) := \left\{
\begin{array}{ll}
q_i & , \quad \text{if } q_i \geqslant \dfrac{\lambda x}{2\alpha_i - \lambda/m} \text{ and } \alpha_i > \dfrac{\lambda}{m}, \\[4mm]
\dfrac{\lambda x}{2\alpha_i - \lambda/m}, & \text{if } q_i < \dfrac{\lambda x}{2\alpha_i - \lambda/m} < Q_i, \\[4mm]
Q_i & , \quad \text{if } Q_i \leqslant \dfrac{\lambda x}{2\alpha_i - \lambda/m} \text{ or } \alpha_i \leqslant \dfrac{\lambda}{m}.
\end{array}
\right.
\\
(5.2c)
\end{array}
$$

Lemma 5.2. *Let* s^* *be an NE of the game* Γ^h_{FL} *and let* $\bar{s}^* = \sum_{i \in [m]} s_i^*/m$. *Then, it holds:*

$$
(5.3) \qquad\qquad s_i^* = \beta_i(\bar{s}^*), \quad \text{for } i \in [m].
$$

As a consequence, $\bar{s}^*$ *satisfies the following fixed-point equation:*

$$
(5.4) \qquad\qquad \bar{s}^* = \frac{1}{m} \sum_{i \in [m]} \beta_i(\bar{s}^*).
$$

Proof. By Lemma 5.1, s^* and $\bar{s}^*$ satisfy (5.1a)-(5.1c). For $i \in [m]$ such that $s_i^* = q_i$, it follows from (5.1a) that

$$
q_i \geqslant \frac{\lambda\left(\bar{s}^* - q_i/m\right)}{2\left(\alpha_i - \lambda/m\right)} \text{ and } \alpha_i > \frac{\lambda}{m},
$$

which is equivalent to (5.2a) by a direct calculation. This implies that $s_i^* = \beta_i(\bar{s}^*)$ for these values of i. By a similar argument, we obtain that $s_i^* = \beta_i(\bar{s}^*)$ for $i \in [m]$ such that $s_i^* = Q_i$. For $i \in [m]$ such that $s_i^* = \lambda \bar{s}^*/(2\alpha_i - \lambda/m)$, from the equality,

$$
\frac{\lambda\left(\bar{s}^* - s_i^*/m\right)}{2\left(\alpha_i - \lambda/m\right)} = \frac{\lambda \bar{s}^*}{2\alpha_i - \lambda/m},
$$

we deduce the equivalence between (5.1b) and (5.2b), which implies $s_i^* = \beta_i(\bar{s}^*)$ in this case. $\qquad\square$

From (5.2a)-(5.2c), we define the following four mappings from $\mathbb{R}_+$ to the subsets of $[m]$:

$$I_1(x) := \left\{ i \in [m] \,\middle|\, q_i \geqslant \frac{\lambda x}{2\alpha_i - \lambda/m} \ \text{ and } \ \alpha_i > \frac{\lambda}{m} \right\};$$

$$I_2(x) := \left\{ i \in [m] \,\middle|\, q_i < \frac{\lambda x}{2\alpha_i - \lambda/m} < Q_i \right\};$$

$$I_3(x) := \left\{ i \in [m] \,\middle|\, Q_i \leqslant \frac{\lambda x}{2\alpha_i - \lambda/m} \ \text{ and } \ \alpha_i > \frac{\lambda}{m} \right\};$$

$$I_4(x) := \left\{ i \in [m] \,\middle|\, \alpha_i \leqslant \frac{\lambda}{m} \right\}.$$

It is straightforward to deduce from the previous definitions that the conditions in (5.2a)-(5.2c) can be replaced by $i \in I_1(x)$, $i \in I_2(x)$, and $i \in I_3(x) \cup I_4(x)$, respectively.

Lemma 5.3. *Fix any $x \in \mathbb{R}_+$. The following statements hold:*

(1) The sets $I_1(x)$, $I_2(x)$, $I_3(x)$, and $I_4(x)$ are mutually disjoint, and

$$(5.5) \qquad\qquad I_1(x) \cup I_2(x) \cup I_3(x) \cup I_4(x) = [m].$$

(2) For any $y \in \mathbb{R}_+$ such that $x \leq y$, we have

$$(5.6) \qquad\qquad I_1(y) \subseteq I_1(x), \quad \text{and} \quad I_3(x) \subseteq I_3(y).$$

Proof. The proof is straightforward by the definitions of I_1, I_2, I_3, and I_4. $\qquad\square$

Proof of Theorem 3.2. Let us prove Theorem 3.2 in the following three steps, where steps 1-2 are for point (1), and steps 3-4 are for points (2a)-(2b).
Step 1 (The case $\lambda \in (0, \bar{\lambda})$ and $\lambda \neq \lambda^*$). Assume that there exist two distinct NEs, denoted by s^1 and s^2. Let $\bar{s}^1 = \sum_{i \in [m]} s_i^1/m$ and $\bar{s}^2 = \sum_{i \in [m]} s_i^2/m$. By (5.3), we have $s_i^1 = \beta_i(\bar{s}^1)$ and $s_i^2 = \beta_i(\bar{s}^2)$ for all $i \in [m]$. Given that $s^1 \neq s^2$, it follows that $\bar{s}^1 \neq \bar{s}^2$. Without loss of generality, assume that $\bar{s}^1 < \bar{s}^2$. By the definition of $\bar{\lambda}$ in (3.2) and the relation $\lambda < \bar{\lambda}$, we have $I_4(\bar{s}_1) = I_4(\bar{s}_2) = \varnothing$. Then, by (5.4), (5.5), and the definition of β_i in (5.2a)-(5.2c), we deduce that

$$(5.7) \qquad m\bar{s}^1 = \sum_{i \in I_1(\bar{s}^1)} q_i + \sum_{i \in I_2(\bar{s}^1)} \frac{\lambda \bar{s}^1}{2\alpha_i - \lambda/m} + \sum_{i \in I_3(\bar{s}^1)} Q_i,$$

$$(5.8) \qquad m\bar{s}^2 = \sum_{i \in I_1(\bar{s}^2)} q_i + \sum_{i \in I_2(\bar{s}^2)} \frac{\lambda \bar{s}^2}{2\alpha_i - \lambda/m} + \sum_{i \in I_3(\bar{s}^2)} Q_i.$$

Since $\bar{s}^1 < \bar{s}^2$, by (5.6), we have $I_1(\bar{s}^2) \subseteq I_1(\bar{s}^1)$, and $I_3(\bar{s}^1) \subseteq I_3(\bar{s}^2)$. Combining with (5.5), we deduce that

$$(5.9) \qquad I_2(\bar{s}^1) \cup I_1(\bar{s}^1) \setminus I_1(\bar{s}^2) = I_2(\bar{s}^2) \cup I_3(\bar{s}^2) \setminus I_3(\bar{s}^1) =: I.$$

By (5.2a) and (5.2c), it holds:

$$(5.10) \qquad\qquad q_i \geqslant \frac{\lambda \bar{s}^1}{2\alpha_i - \lambda/m}, \quad \text{for } i \in I_1(\bar{s}^1) \setminus I_1(\bar{s}^2),$$

$$(5.11) \qquad\qquad Q_i \leqslant \frac{\lambda \bar{s}^2}{2\alpha_i - \lambda/m}, \quad \text{for } i \in I_3(\bar{s}^2) \setminus I_3(\bar{s}^1).$$

Substituting (5.10) to (5.7) and (5.11) to (5.8), and applying (5.9) , we obtain that

$$
(5.12) \qquad m\bar{s}^1 \geqslant \sum_{i \in I_1(\bar{s}^2)} q_i + \sum_{i \in I} \frac{\lambda \bar{s}^1}{2\alpha_i - \lambda/m} + \sum_{i \in I_3(\bar{s}^1)} Q_i,
$$

$$
(5.13) \qquad m\bar{s}^2 \leqslant \sum_{i \in I_1(\bar{s}^2)} q_i + \sum_{i \in I} \frac{\lambda \bar{s}^2}{2\alpha_i - \lambda/m} + \sum_{i \in I_3(\bar{s}^1)} Q_i.
$$

Taking the difference between (5.13) and (5.12), and noting that $\bar{s}^1 < \bar{s}^2$, we have

$$
(5.14) \qquad 1 \leqslant \sum_{i \in I} \frac{\lambda}{2m\alpha_i - \lambda}.
$$

On the other hand, by (5.12), if $I_1(\bar{s}^2) \cup I_3(\bar{s}^1) \neq \varnothing$, then

$$
1 > \sum_{i \in I} \frac{\lambda}{2m\alpha_i - \lambda}.
$$

We obtain a contradiction with (5.14). Therefore, $I_1(\bar{s}^2) \cup I_3(\bar{s}^1) = \varnothing$, giving $I = [m]$. As a consequence, by (5.12) and (5.14), we obtain that

$$
1 = \sum_{i \in [m]} \frac{\lambda}{2m\alpha_i - \lambda},
$$

which is a contradiction with the assumption $\lambda \neq \lambda^*$ (recall the definition of λ^* from (3.3)). We conclude the uniqueness of the NE in this case.

Step 2 (The case $\lambda \geqslant \bar{\lambda}$). Assume that s^1 and s^2 are two distinct NEs and $\bar{s}^1 < \bar{s}^2$. Analogous to (5.7)-(5.8) and noting that $\lambda \geqslant \bar{\lambda}$, we deduce that

$$
m\bar{s}^1 = \sum_{i \in I_1(\bar{s}^1)} q_i + \sum_{i \in I_2(\bar{s}^1)} \frac{\lambda \bar{s}^1}{2\alpha_i - \lambda/m} + \sum_{i \in I_3(\bar{s}^1)} Q_i + \sum_{i \in I_4(\bar{s}^1)} Q_i,
$$

$$
m\bar{s}^2 = \sum_{i \in I_1(\bar{s}^2)} q_i + \sum_{i \in I_2(\bar{s}^2)} \frac{\lambda \bar{s}^2}{2\alpha_i - \lambda/m} + \sum_{i \in I_3(\bar{s}^2)} Q_i + \sum_{i \in I_4(\bar{s}^2)} Q_i.
$$

Noting that $I_4(x)$ is independent of x, we have $I_4(\bar{s}^1) = I_4(\bar{s}^2) \neq \varnothing$ (since $\lambda \geqslant \bar{\lambda}$). By the same reasoning as in step 1, we obtain (5.14), i.e.,

$$
1 \leqslant \sum_{i \in I} \frac{\lambda}{2m\alpha_i - \lambda}.
$$

Similar to (5.12), we have

$$
m\bar{s}^1 \geqslant \sum_{i \in I_1(\bar{s}^2)} q_i + \sum_{i \in I} \frac{\lambda \bar{s}^1}{2\alpha_i - \lambda/m} + \sum_{i \in I_3(\bar{s}^1)} Q_i + \sum_{i \in I_4(\bar{s}^1)} Q_i
$$

$$
\geqslant \sum_{i \in I} \frac{\lambda \bar{s}^1}{2\alpha_i - \lambda/m} + \sum_{i \in I_4(\bar{s}^1)} Q_i
$$

$$
> \sum_{i \in I} \frac{\lambda \bar{s}^1}{2\alpha_i - \lambda/m},
$$

where the last inequality is strict since $I_4(\bar{s}^1)$ is nonempty. We obtain a contradiction with (5.14), which implies the uniqueness of NE in this case.

Step 3 (The case $\lambda = \lambda^*$ and $c_1 \geqslant c_2$). Assume that s^1 and s^2 are two distinct NEs. By the proof in step 1, we deduce that $I_1(\bar{s}^2) \cup I_3(\bar{s}^1) = \emptyset$. Applying the relation $I_3(\bar{s}^1) = \emptyset$ to (5.7) and noting that λ^* is the solution of (3.3), we obtain that

$$\sum_{i \in I_1(\bar{s}^1)} \frac{\lambda^* \bar{s}^1}{2\alpha_i - \lambda^*/m} = \sum_{i \in I_1(\bar{s}^1)} q_i.$$

Combining with the definition of $I_1(\bar{s}^1)$, it follows that

$$\frac{\lambda^* \bar{s}^1}{2\alpha_i - \lambda^*/m} = q_i, \quad \text{for } i \in I_1(\bar{s}^1).$$

Given the fact that $I_3(\bar{s}^1) = \emptyset$, we deduce that

$$q_i \leqslant \frac{\lambda^* \bar{s}^1}{2\alpha_i - \lambda^*/m} < Q_i, \quad \text{for } i \in [m].$$

This is equivalent to

$$\max_{i \in [m]} \left\{ \frac{2\alpha_i q_i}{\lambda^*} - \frac{q_i}{m} \right\} \leqslant \bar{s}^1, \quad \text{and} \quad \bar{s}^1 < \min_{i \in [m]} \left\{ \frac{2\alpha_i Q_i}{\lambda^*} - \frac{Q_i}{m} \right\}.$$

Recalling the formula of c_1 and c_2 from (3.4), by the previous inequality, $c_1 \leq \bar{s}^1 < c_2$, which is a contradiction with our assumption $c_1 \geqslant c_2$. The uniqueness of the NE follows.

Step 4 (The case $\lambda = \lambda^*$ and $c_1 < c_2$). Let us take some $s^* \in \mathcal{S}^h$ by (3.5), i.e., for some $c \in (c_1, c_2)$,

$$s_i^* = \frac{\lambda^* c}{2\alpha_i - \lambda^*/m}, \quad \forall i \in [m].$$

It follows that

$$\bar{s}^* = \frac{1}{m} \sum_{i \in [m]} s_i^* = c \in (c_1, c_2).$$

By direct calculation, we obtain that

$$\frac{\lambda^* (\bar{s}^* - s_i^*/m)}{2 (\alpha_i - \lambda^*/m)} = \frac{c\lambda^*}{2\alpha_i - \lambda^*/m}, \quad \text{for } i \in [m].$$

Since $c \in (c_1, c_2)$, it follows that

$$q_i < \frac{\lambda^* (\bar{s}^* - s_i^*/m)}{2 (\alpha_i - \lambda^*/m)} < Q_i, \quad \text{for } i \in [m].$$

Combining with (3.3), we deduce that s^* satisfies the fixed-point system (5.1a)-(5.1c), thus it is an NE. This implies the sufficiency of condition (3.5).

Since the choice of c is infinite, we deduce that the game Γ_{FL}^h has infinite NEs. In particular, the game Γ_{FL}^h has two distinct NEs. Then, by a similar argument of step 3, we deduce that for any NE of the game Γ_{FL}^h, denoted by s', it holds:

$$(5.15) \qquad\qquad q_i \leqslant \frac{\lambda^* \bar{s}'}{2\alpha_i - \lambda^*/m} \leqslant Q_i, \quad \text{for } i \in [m],$$

where $\bar{s}' = \sum_{i \in [m]} s_i'/m$. In addition, we deduce that $\bar{s}' \in (c_1, c_2)$ from (5.15) and (3.4). Moreover, by (5.3), $s_i' = \beta_i(\bar{s}')$ for all i. Combining with (5.15), we obtain that

$$s_i' = \frac{\lambda^* \bar{s}'}{2\alpha_i - \lambda^*/m} \quad \text{for } i \in [m].$$

The necessity of condition (3.5) follows. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\square$

Proof of Corollary 3.5. We prove Corollary 3.5 in the following three steps:

Step 1 (Proof of points (1) and (4)). Assume that $\lambda \in (0, \lambda_1)$. Since $\lambda_1 \leq \lambda^*$, by Theorem 3.2, the NE is unique. Therefore, it suffices to prove that $(q_i)_{i=1}^m$ is an NE. Let $\bar{q} = \sum_{i \in [m]} q_i/m$. Recalling the formula of λ_1 from (3.7), we deduce that

$$(5.16) \qquad q_i \geq \frac{\lambda(\bar{q} - q_i/m)}{2(\alpha_i - \lambda/m)}, \quad \text{for } i \in [m].$$

Therefore, $(q_i)_{i=1}^m$ is an NE since it satisfies the fixed-point system (5.1a)-(5.1c). The case where $\lambda \in (\lambda_2, +\infty)$ is proved by a similar argument, using the definition of λ_2 from (3.7)

Step 2 (Proof of point (2)). Assume that $\lambda < \lambda^*$. Let s^* be the unique NE, and $\bar{s}^* = \sum_{i \in [m]} s_i^*/m$. Since $\lambda^* < \bar{\lambda}$, we have $I_4(\bar{s}^*) = \emptyset$. By (5.4), we obtain that

$$m\bar{s}^* = \sum_{i \in I_1(\bar{s}^*)} q_i + \sum_{i \in I_2(\bar{s}^*)} \frac{\lambda\bar{s}^*}{2\alpha_i - \lambda/m} + \sum_{i \in I_3(\bar{s}^*)} Q_i.$$

If $I_1(\bar{s}^*) = \emptyset$, then it follows that

$$m\bar{s}^* = \sum_{i \in I_2(\bar{s}^*)} \frac{\lambda\bar{s}^*}{2\alpha_i - \lambda/m} + \sum_{i \in I_3(\bar{s}^*)} Q_i \leq \sum_{i \in [m]} \frac{\lambda\bar{s}^*}{2\alpha_i - \lambda/m}.$$

This implies that

$$1 \leq \sum_{i \in [m]} \frac{\lambda}{2\alpha_i - \lambda/m},$$

which is a contradiction with the assumption $\lambda < \lambda^*$. Therefore, $I_1(\bar{s}^*) \neq \emptyset$. This implies that

$$m\bar{s}^* \leq \sum_{i \in I_1(\bar{s}^*)} q_i + \sum_{i \in [m]\setminus I_1(\bar{s}^*)} \frac{\lambda\bar{s}^*}{2\alpha_i - \lambda/m} \leq \sum_{i \in I_1(\bar{s}^*)} q_i + \sum_{i \in [m]\setminus I_1(\bar{s}^*)} \frac{\lambda^*\bar{s}^*}{2\alpha_i - \lambda^*/m}.$$

Suppose that

$$(5.17) \qquad \bar{s}^* > c_1 = \max_{i \in [m]} \left\{ \frac{2\alpha_i q_i}{\lambda^*} - \frac{q_i}{m} \right\}.$$

Then,

$$1 < \sum_{i \in I_1(\bar{s}^*)} \frac{q_i}{m\left(\frac{2\alpha_i q_i}{\lambda^*} - \frac{q_i}{m}\right)} + \sum_{i \in [m]\setminus I_1(\bar{s}^*)} \frac{\lambda^*}{2m\alpha_i - \lambda^*} = \sum_{i \in [m]} \frac{\lambda^*}{2m\alpha_i - \lambda^*} = 1,$$

where the first inequality is strict since $I_1(\bar{s}^*)$ is non-empty. This leads to a contradiction. Therefore, (5.17) cannot hold in this case, implying that $\bar{s}^* \leq c_1$.

Step 3 (Proof of point (3)). Assume that $\lambda > \lambda^*$. Let s^* be the unique NE, and $\bar{s}^* = \sum_{i \in [m]} s_i^*/m$. It follows that

$$m\bar{s}^* = \sum_{i \in I_1(\bar{s}^*)} q_i + \sum_{i \in I_2(\bar{s}^*)} \frac{\lambda\bar{s}^*}{2\alpha_i - \lambda/m} + \sum_{i \in I_3(\bar{s}^*)} Q_i + \sum_{i \in I_4(\bar{s}^*)} Q_i.$$

If $I_4(\bar{s}^*) = \emptyset$, we deduce that $I_3(\bar{s}^*) \neq \emptyset$ by a similar argument in step 2. Then, following step 2, we deduce that $\bar{s}^* \geq c_2$.

Now assuming that $I_3(\bar{s}^*) = \emptyset$ and $I_4(\bar{s}^*) \neq \emptyset$, then

$$m\bar{s}^* \geq \sum_{i \in I_1(\bar{s}^*)\cup I_2(\bar{s}^*)} \frac{\lambda\bar{s}^*}{2\alpha_i - \lambda/m} + \sum_{i \in I_4(\bar{s}^*)} Q_i = \sum_{i \in [m]\setminus I_4(\bar{s}^*)} \frac{\lambda\bar{s}^*}{2\alpha_i - \lambda/m} + \sum_{i \in I_4(\bar{s}^*)} Q_i.$$

Since $2\alpha_i - \lambda/m > 0$ for $i \notin I_4(\bar{s}^*)$, we deduce, by the monotonicity of $\lambda/(2m\alpha_i - \lambda)$, that

$$
(5.18) \qquad 1 \geqslant \sum_{i \in [m] \setminus I_4(\bar{s}^*)} \frac{\lambda^*}{2m\alpha_i - \lambda^*} + \sum_{i \in I_4(\bar{s}^*)} \frac{Q_i}{m\bar{s}^*}.
$$

Suppose that

$$
(5.19) \qquad \bar{s}^* < c_2 = \min_{i \in [m]} \left\{ \frac{2\alpha_i Q_i}{\lambda^*} - \frac{Q_i}{m} \right\}.
$$

Substituting (5.19) into (5.18), we obtain that

$$
1 > \sum_{i \in [m] \setminus I_4(\bar{s}^*)} \frac{\lambda^*}{2m\alpha_i - \lambda^*} + \sum_{i \in I_4(\bar{s}^*)} \frac{Q_i}{m \left(\frac{2\alpha_i Q_i}{\lambda^*} - \frac{Q_i}{m} \right)} = \sum_{i \in [m]} \frac{\lambda^*}{2m\alpha_i - \lambda^*} = 1.
$$

This leads to a contradiction. Therefore, (5.19) cannot hold in this case, implying that $\bar{s}^* \geqslant c_2$. $\qquad\square$

5.2. **Convergence of the best-response algorithm for potential games.** This subsection is dedicated to proving a general convergence result (see Theorem 5.6) of Algorithm 1 when applied to a (weighted) potential game Γ (with adaptive modifications in Algorithm 1 to fit Γ). This subsection is independent of the FL context discussed elsewhere in this article. To simplify the presentation, we mention that notations used in this subsection are independent of those in other sections.

Consider a (weighted) potential game Γ (see Definition 1.3) with strategy sets $\mathcal{S}_i \subseteq \mathbb{R}^d$, payoff functions $P_i(s_i, s-i)$, and a w-potential P, where $w_i > 0$ for all $i \in [m]$. To prove the convergence of Algorithm 1, we need Assumption 5.4 on the convexity and regularity of $\mathcal{S}_i$ and P_i. The proof of Theorem 5.6 is inspired by the convergence of the block coordinate descent algorithm for multi-convex functions and congestion games, see [39] and [21, Sec. 3.2].

Assumption 5.4. *The following holds:*
   (1) The strategy sets $\mathcal{S}_i$'s are compact and convex subsets of $\mathbb{R}^d$.
   (2) For any $i \in [m]$, there exists a constant $\alpha_i > 0$ such that, the payoff function $P_i(s_i, s_{-i})$ is α_i-strongly concave with respect to s_i for any $s_{-i} \in \mathcal{S}_{-i}$, and $\alpha_{min} = \min_{i \in [m]} \{\alpha_i\}$.
   (3) For any $i \in [m]$, the payoff function $P_i(s_i, s-i)$ is continuously differentiable with respect to s_i, and this partial gradient is denoted by $\nabla_i P_i(s_i, s_{-i})$.
   (4) There exists a constant $L > 0$ such that for any $i \in [m]$, $\nabla_i P_i(s_i, s_{-i})$ is L-Lipschitz with respect to (s_i, s_{-i}).

Before the presentation of the general convergence result of the best-response algorithm, we recall the definition of the ϵ-Nash equilibrium.

Definition 5.5 (ϵ-NE). In a game Γ, for any $\epsilon \geqslant 0$, we call a point $s^\epsilon \in \mathcal{S}$ an ϵ-NE if the following inequality holds:

$$
P_i\left(s_i^\epsilon, s_{-i}^\epsilon\right) \geqslant P_i\left(s_i, s_{-i}^\epsilon\right) - \epsilon, \quad \forall s_i \in \mathcal{S}_i, \forall i \in [m].
$$

In particular, if $\epsilon = 0$, then s^0 is an NE.

Theorem 5.6. *Let Γ be a w-potential game and let Assumption 5.4 hold true. Let s^k be the result of Algorithm 1 at k-th iteration, for any $k \geqslant 1$. We have the following assertions:*
   (1) For any $K \geqslant 1$, there exists $k \in \{1, \ldots, K\}$ such that s^k is an $\mathcal{O}(1/K)$-NE.
   (2) Any cluster point of the sequence $\{s^k\}_{k \geqslant 1}$ is an NE of Γ.

Proof. Let us prove the assertions in the following steps.
Step 1 (Finite sum of square). Let us define $p_i^k \colon \mathcal{S}_i \to \mathbb{R}$,

$$
p_i^k(s_i) = P_i(s_1^k, \ldots, s_{i-1}^k, s_i, s_{i+1}^{k-1}, \ldots, s_m^{k-1}).
$$

Let P be a w-potential of Γ. Then, for any i,

$$\frac{1}{w_i}\left(p_i^k(s_i^k) - p_i^k(s_i^{k-1})\right) = P(s_1^k, \ldots, s_{i-1}^k, s_i^k, s_{i+1}^{k-1}, \ldots, s_m^{k-1})$$
$$- P(s_1^k, \ldots, s_{i-1}^k, s_i^{k-1}, s_{i+1}^{k-1}, \ldots, s_m^{k-1}).$$

Summing the previous equality over i, we obtain that

$$\sum_{i\in[m]} \frac{1}{w_i}\left(p_i^k(s_i^k) - p_i^k(s_i^{k-1})\right) = P(s^k) - P(s^{k-1}).$$

On the other hand, by the strong concavity of P_i w.r.t. s_i and the fact that s_i^k is the maximizer of p_i^k, we deduce that

$$p_i^k(s_i^k) - p_i^k(s_i^{k-1}) \geqslant \frac{\alpha_i}{2}\|s_i^k - s_i^{k-1}\|^2.$$

It follows that

$$\sum_{i=1}^m \frac{\alpha_i}{2w_i}\|s_i^k - s_i^{k-1}\|^2 \leqslant P(s^k) - P(s^{k-1}).$$

Summing the previous inequality over k from 1 to K, we deduce that

$$\sum_{k=1}^K \|s^k - s^{k-1}\|^2 \leqslant \frac{2}{\delta}\left(P(s^{K+1}) - P(s^0)\right) \leqslant \frac{2}{\delta}\left(P^* - P(s^0)\right),$$

where $\delta = \min_{i\in[m]}\{\alpha_i/w_i\} > 0$, and P^* is the maximum value of P and is finite.

Step 2 (Proof of point (1)). Let us assume that $\|s^k - s^{k-1}\| \leqslant \epsilon$ for some $k \geqslant 1$ and $\epsilon > 0$. Then, by the strong concavity of P_i w.r.t. s_i, we have

$$P_i(s_i, s_{-i}^k) - P_i(s_i^k, s_{-i}^k) \leqslant \langle \nabla_i P_i(s_i^k, s_{-i}^k), s_i - s_i^k\rangle - \frac{\alpha_{\min}}{2}\|s_i - s_i^k\|^2.$$

We decompose the linear term on the right-hand-side of the previous inequality in the following way:

$$\langle \nabla_i P_i(s_i^k, s_{-i}^k), s_i - s_i^k\rangle = \gamma_1 + \gamma_2,$$

where

$$\gamma_1 = \langle \nabla_i P_i(s_i^k, s_{-i}^k) - \nabla p_i^k(s_i^k), s_i - s_i^k\rangle,$$
$$\gamma_2 = \langle \nabla p_i^k(s_i^k), s_i - s_i^k\rangle.$$

Since s_i^k is the maximizer of p_i^k, the first-order optimality condition implies that $\gamma_2 \leqslant 0$. Besides, by the definition of p_i^k, we have

$$\nabla p_i^k(s_i^k) = \nabla_i P_i(s_1^k, \ldots, s_{i-1}^k, s_i^k, s_{i+1}^{k-1}, \ldots, s_m^{k-1}).$$

By the Lipschitz continuity of $\nabla_i P_i$, we deduce that

$$\|\nabla_i P_i(s_i^k, s_{-i}^k) - \nabla p_i^k(s_i^k)\| \leqslant L\|s^k - (s_1^k, \ldots, s_{i-1}^k, s_i^k, s_{i+1}^{k-1}, \ldots, s_m^{k-1})\| \leqslant L\|s^k - s^{k-1}\| \leqslant L\epsilon.$$

Therefore, for any i and $s_i \in \mathcal{S}_i$,

$$P_i(s_i, s_{-i}^k) - P_i(s_i^k, s_{-i}^k) \leqslant L\epsilon\|s_i - s_i^k\| - \frac{\alpha_{\min}}{2}\|s_i - s_i^k\|^2 \leqslant \frac{L^2\epsilon^2}{2\alpha_{\min}}.$$

Therefore, we prove that s^k is an $(L^2\epsilon^2/2\alpha_{\min})$-NE. From step 1, fixing any $K \geqslant 1$, there exists $k_0 \leqslant K$ such that

$$\|s^{k_0} - s^{k_0-1}\|^2 \leqslant \frac{2}{\delta K}(P^* - P(s^0)),$$

where P^* is the maximum value of P over $\mathcal{S}$ and is finite. Let $k = k_0$, it follows that s^{k_0} is an ϵ'-NE, where

$$\epsilon' = \frac{L^2}{2\alpha_{\min}} \frac{2}{\delta K}(P^* - P(s^0)) = \frac{L^2(P^* - P(s^0))}{\delta\,\alpha_{\min}} \frac{1}{K}.$$

Step 3 (Proof of point (2)). By the compactness of $\mathcal{S}$, assume that a subsequence of $\{s^k\}_{k\geqslant 1}$, denoted by $\{s^{\varphi(k)}\}_{k\geqslant 1}$, converges to some point $s^* \in \mathcal{S}$. By step 1, we have that $\|s^{\varphi(k)-1} - s^{\varphi(k)}\| \to 0$. Similar to step 2, we have

$$P_i(s_i, s_{-i}^*) - P_i(s_i^*, s_{-i}^*) \leqslant \langle \nabla_i P_i(s_i^*, s_{-i}^*), s_i - s_i^* \rangle - \frac{\alpha_{\min}}{2}\|s_i - s_i^*\|^2.$$

The previous first-order term is decomposed by

$$
\begin{aligned}
\langle \nabla_i P_i(s_i^*, s_{-i}^*), s_i - s_i^* \rangle &= \langle \nabla_i P_i(s_i^{\varphi(k)}, s_{-i}^{\varphi(k)}), s_i - s_i^{\varphi(k)} \rangle + \langle \nabla_i P_i(s_i^*, s_{-i}^*), s_i^{\varphi(k)} - s_i^* \rangle \\
&\quad + \langle \nabla_i P_i(s_i^*, s_{-i}^*) - \nabla_i P_i(s_i^{\varphi(k)}, s_{-i}^{\varphi(k)}), s_i - s_i^{\varphi(k)} \rangle \\
&\leqslant L\left(\|s^{\varphi(k)} - s^{\varphi(k)-1}\| + \|s^* - s^{\varphi(k)}\|\right)\|s_i - s_i^{\varphi(k)}\| \\
&\quad + \|\nabla_i P_i(s^*)\|\|s_i^{\varphi(k)} - s_i^*\|,
\end{aligned}
$$

where the last inequality is deduced from a similar argument of step 2 and the Lipschitz continuity of $\nabla_i P_i$. By passing k to infinity, we deduce that $P_i(s_i, s_{-i}^*) - P_i(s_i^*, s_{-i}^*) \leqslant 0$ for any i and s_i. Therefore, s^* is an NE. $\qquad\square$

6. Conclusions and future work

In this paper, we introduce a potential game framework to model the federated learning (FL) training-cost-incentive process. We prove the existence of Nash equilibria (NEs) in the proposed game and further investigate the uniqueness of the NE in a homogeneous scenario where clients maintain constant efforts throughout the training process. Additionally, we characterize the evolution of NEs in relation to the server's reward factor, identifying three critical thresholds: the activation point, the jump point, and the saturation point. Notably, our findings emphasize the significance of the jump point, where a sharp increase in clients' training efforts is observed, indicating it as the optimal reward factor for the server. Meanwhile, extensive numerical experiments strongly support our theoretical results. Overall, our work provides a robust foundation for understanding the interplay between clients' training efforts and the server's incentives.

For future work, we propose the following perspectives.

(1) A first extension can focus on more general heterogeneous scenarios. For instance, it is reasonable to incorporate a discount factor into the server's reward scheme based on the number of rounds. This aligns with the convergence properties of widely used stochastic gradient algorithms in machine learning, where significant gains in model performance are typically observed during the early training stages [4, Sec. 4]. This phenomenon is also reflected in our simulation results shown in Figure 4.3, where major improvements are made in the first several rounds. Furthermore, adding a discount factor may also help address the issue of non-uniqueness of the NE in heterogeneous cases, as the counterexample mentioned in Remark 3.10 no longer applies in this scenario.

(2) A second perspective focuses on the asymptotic behavior of our FL game as the number of clients m approaches infinity. In this scenario, a nonatomic version of Γ_{FL} or Γ_{FL}^h will be analyzed. The NEs in this setting are characterized by an optimality condition over the distribution of training efforts among a continuum of clients, see [7, Thm. 1]. It is worth noting that proving the uniqueness of NE in the nonatomic case is generally challenging when the price increases with respect to the aggregate, as discussed in [7, Prop. 3]. Our

techniques used in proving the uniqueness results in Section 5 can be naturally adapted to these nonatomic games.

(3) Another promising direction is to explore more complex and practical reward allocation schemes in the FL game formulation. In our current work, after determining the unit price, each client's revenue is linearly correlated with its effort. However, the true relationship between a client's training effort and the resulting improvement in model performance is unknown and difficult to express analytically. Some attempts have been made to evaluate each client's contribution (see [23, 40]). However, incorporating these evaluations into the game formulation and analysis remains a challenging problem.

Acknowledgments

References

[1] R. J. Aumann. Backward induction and common knowledge of rationality. *Games and Economic Behavior*, 8(1):6–19, 1995.

[2] S. Banabilah, M. Aloqaily, E. Alsayed, N. Malik, and Y. Jararweh. Federated learning review: Fundamentals, enabling technologies, and future applications. *Information processing & management*, 59(6):103061, 2022.

[3] E. Barron. *Game Theory: An Introduction*. Wiley, 1 edition, 2013.

[4] L. Bottou, F. E. Curtis, and J. Nocedal. Optimization methods for large-scale machine learning. *SIAM Review*, 60(2):223–311, 2018.

[5] A. Britzelmeier, A. Dreves, and M. Gerdts. Numerical solution of potential games arising in the control of cooperative automatic vehicles. In *2019 Proceedings of the Conference on Control and Its Applications*, pages 38–45. SIAM, 2019.

[6] P. Cardaliaguet. Notes on mean field games. Technical report, 2010.

[7] M.-W. Cheung and R. Lahkar. Nonatomic potential games: The continuous strategy case. *Games and Economic Behavior*, 108:341–362, 2018.

[8] A. Cournot. The mathematical principles of the theory of wealth (translated from recherches sur les principes mathématiques de la théorie des richesses, by nt bacon), 1927.

[9] L. Dong and Y. Zhang. Federated learning service market: A game theoretic analysis. In *2020 International Conference on Wireless Communications and Signal Processing (WCSP)*, pages 227–232, 2020.

[10] S. Durand and B. Gaujal. Complexity and optimality of the best response algorithm in random potential games. In *International Symposium on Algorithmic Game Theory*, pages 40–51. Springer, 2016.

[11] 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.

[12] Y. He, M. Luo, B. Wu, L. Sun, Y. Wu, Z. Liu, and K. Xiao. A game theory-based incentive mechanism for collaborative security of federated learning in energy blockchain environment. *IEEE Internet of Things Journal*, pages 1–1, 2023.

[13] M. K. Jensen. Aggregative games and best-reply potentials. *Economic Theory*, 43(1):45–66, 2010.

[14] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

[15] M. L. Katz and C. Shapiro. Network externalities, competition, and compatibility. *The American Economic Review*, 75(3):424–440, 1985.

[16] 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*, 58(10):88–93, 2020.

[17] D. M. Kreps and J. A. Scheinkman. Quantity precommitment and bertrand competition yield cournot outcomes. *The Bell Journal of Economics*, pages 326–337, 1983.

[18] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[19] 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*, 7(9):8430–8439, 2020.

[20] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems*, 2:429–450, 2020.

[21] K. Liu, N. Oudjane, and C. Wan. Approximate nash equilibria in large nonconvex aggregative games. *Mathematics of Operations Research*, 48(3):1791–1809, 2023.

[22] T. Liu, J. Wang, X. Zhang, and D. Cheng. Game theoretic control of multiagent systems. *SIAM Journal on Control and Optimization*, 57(3):1691–1709, 2019.

[23] Z. Liu, Y. Chen, H. Yu, Y. Liu, and L. Cui. Gtg-shapley: Efficient and accurate participant contribution evaluation in federated learning. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 13(4):1–21, 2022.

[24] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, pages 1273–1282. PMLR, 2017.

[25] D. Monderer and L. S. Shapley. Potential games. *Games and Economic Behavior*, 14(1):124–143, 1996.

[26] J. F. Nash Jr. Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences*, 36(1):48–49, 1950.

[27] M. J. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press, Cambridge, Mass, 1994.

[28] A. Raschellà, F. Bouhafs, M. Mackay, Q. Shi, J. Ortín, J. R. Gállego, and M. Canales. A dynamic access point allocation algorithm for dense wireless lans using potential game. *Computer Networks*, 167:106991, 2020.

[29] H. Ren, J. Deng, and X. Xie. Grnn: Generative regression neural network—a data leakage attack for federated learning. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 13(4):1–24, 2022.

[30] J. B. Rosen. Existence and uniqueness of equilibrium points for concave n-person games. *Econometrica*, 33(3):520–534, 1965.

[31] R. Selten. *Preispolitik Der Mehrproduktenunternehmung in Der Statischen Theorie*, volume 16 of *Ökonometrie Und Unternehmensforschung / Econometrics and Operations Research*. Springer, Berlin, Heidelberg, 1970.

[32] Y. Song, Z. Wang, and E. Zuazua. Fedadmm-insa: An inexact and self-adaptive admm for federated learning. *Neural Networks*, 181:106772, 2025.

[33] B. Swenson, R. Murray, and S. Kar. On best-response dynamics in potential games. *SIAM Journal on Control and Optimization*, 56(4):2734–2767, 2018.

[34] F. Szidarovszky and S. Yakowitz. A new proof of the existence and uniqueness of the cournot equilibrium. *International Economic Review*, pages 787–789, 1977.

[35] E. Tahanian, M. Amouei, H. Fateh, and M. Rezvani. A game-theoretic approach for robust federated learning. *International Journal of Engineering*, 34(4):832–842, 2021.

[36] X. Tu, K. Zhu, N. C. Luong, D. Niyato, Y. Zhang, and J. Li. Incentive mechanisms for federated learning: From economic and game theoretic perspective. *IEEE Transactions on Cognitive Communications and Networking*, 8(3):1566–1593, 2022.

[37] H. Von Stackelberg. *Marktform und Gleichgewicht*. J. Springer, 1934.

[38] H. Wu and H. Shang. Potential game for dynamic task allocation in multi-agent system. *ISA Transactions*, 102:208–220, 2020.

[39] Y. Xu and W. Yin. A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on Imaging Sciences*, 6(3):1758–1789, 2013.

[40] Y. Xue, C. Niu, Z. Zheng, S. Tang, C. Lyu, F. Wu, and G. Chen. Toward understanding the influence of individual clients in federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 10560–10567, 2021.

[41] L. Zhang, T. Zhu, P. Xiong, W. Zhou, and P. S. Yu. A robust game-theoretical federated learning framework with joint differential privacy. *IEEE Transactions on Knowledge and Data Engineering*, 35(4):3333–3346, 2023.

[42] X. Zhang, M. Hong, and N. Elia. Understanding a class of decentralized and federated optimization algorithms: A multirate feedback control perspective. *SIAM Journal on Optimization*, 33(2):652–683, 2023.

[43] L. Zhu, Z. Liu, and S. Han. Deep leakage from gradients. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.