



Trinity College Dublin

Coláiste na Tríonóide, Baile Átha Cliath

The University of Dublin

A Critical Learning Period-Aware Incentive Mechanism for Federated Learning

IEEE VTC2025-Spring

Thanh Linh Nguyen, Quoc-Viet Pham

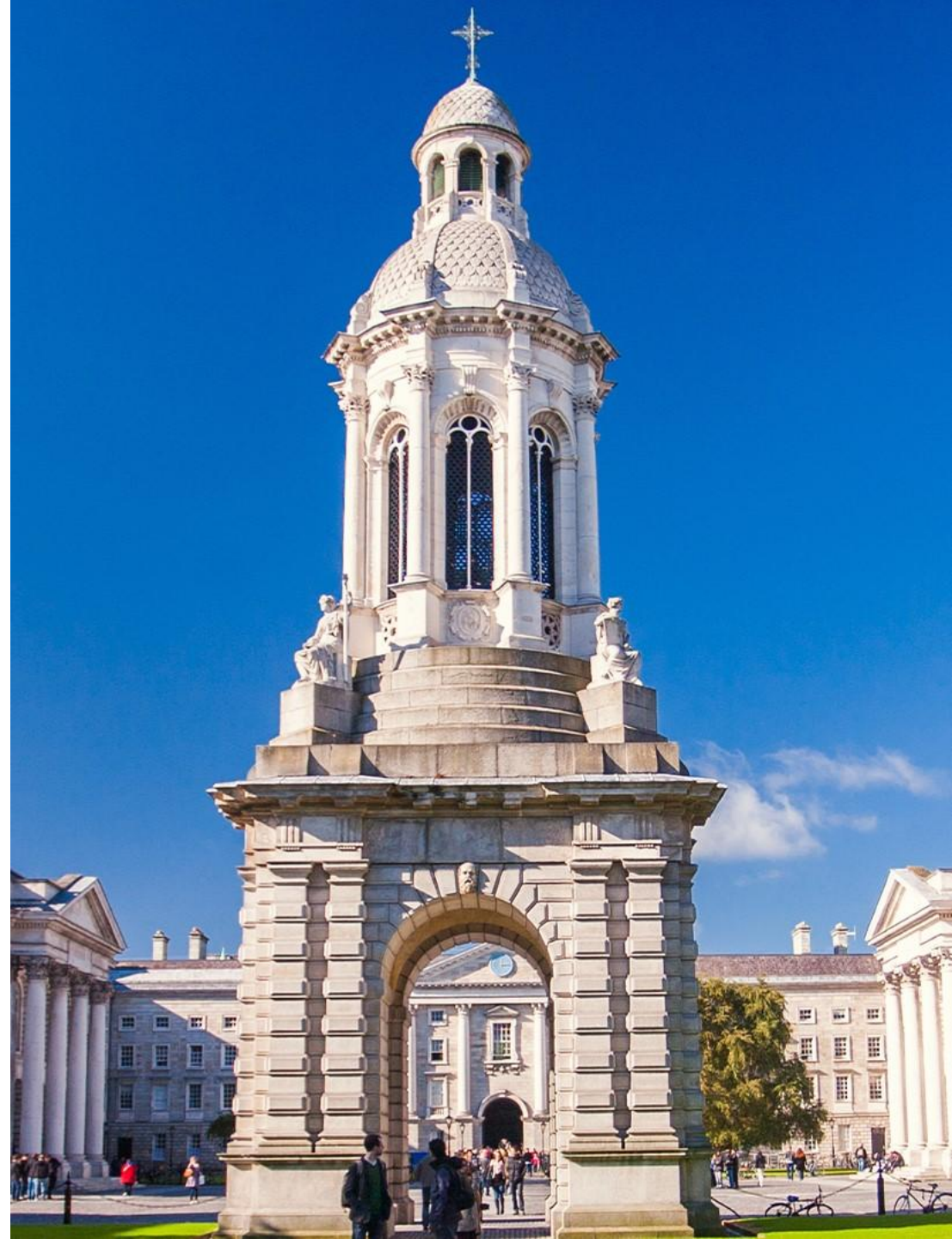


ADVANCECART



Taighde Éireann
Research Ireland

Oslo, June 2025



Background: Federated Learning (FL)

❖ Definition

- A privacy-preserving data sharing/learning paradigm [1]

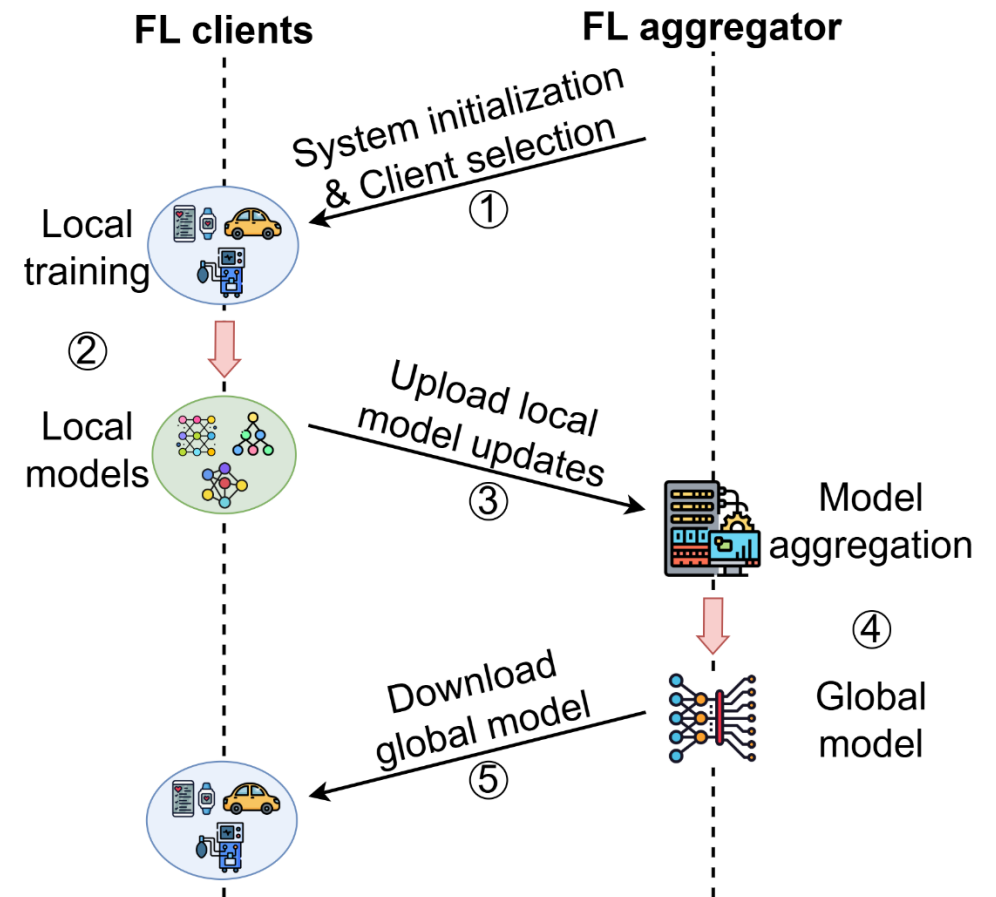
❖ Components

- Distributed FL clients (e.g., IoT devices, organizations,...)
- A central server (i.e., FL aggregator)

❖ Workflow

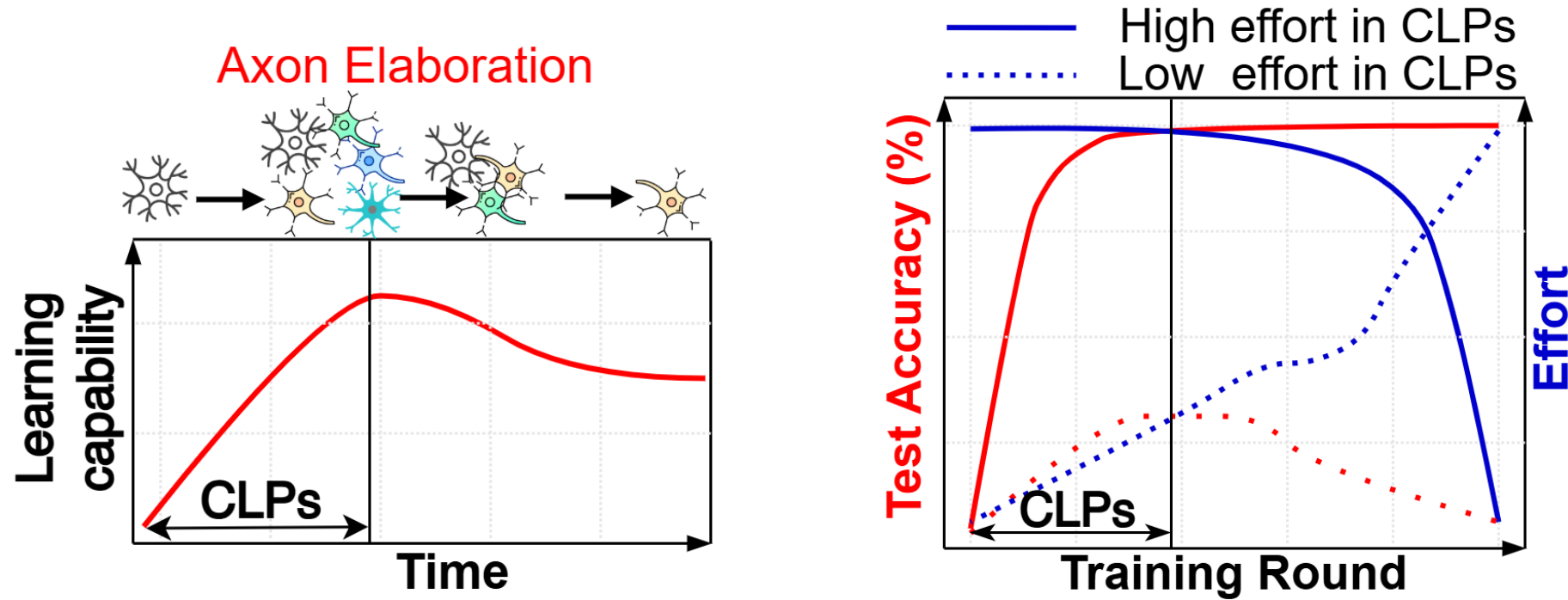
- Instead of sharing raw data, FL only share the trained model parameters.

Data Stays, Intelligence Unites



[1]. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Ar- cas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial Intelligence and Statistics*, 2017, pp. 1273–1282.

Background: Critical Learning Periods (CLPs)



➔ The importance of each training round is not equally important in FL [1-4].

- [1]. Michael Kleinman and Alessandro Achille and Stefano Soatto, "Critical Learning Periods Emerge Even in Deep Linear Networks", *ICLR* (spotlight, 2024)
- [2]. Gang Yan, Hao Wang, Xu Yuan, and Jian Li, "CriticalFL: A Critical Learning Periods Augmented Client Selection Framework for Efficient Federated Learning", *ACM SIGKDD*, 2023.
- [3]. Achille, Alessandro, Matteo Rovere, and Stefano Soatto. "Critical learning periods in deep neural networks.", *ICLR* (2017).
- [4]. E. R. Kandel, J. H. Schwartz, T. M. Jessell, S. Siegelbaum, A. J. Hudspeth, S. Mack et al., "Principles of neural science. McGraw-hill New York", 2000.

Motivation

■ #Research Gaps and Challenges:

- Assumption of voluntary participation among FL clients
- Information asymmetry between clients and the central server
- Equal treatment of client contributions across training rounds
- Centralized incentive coordination and disputes due to the reliance on the central server

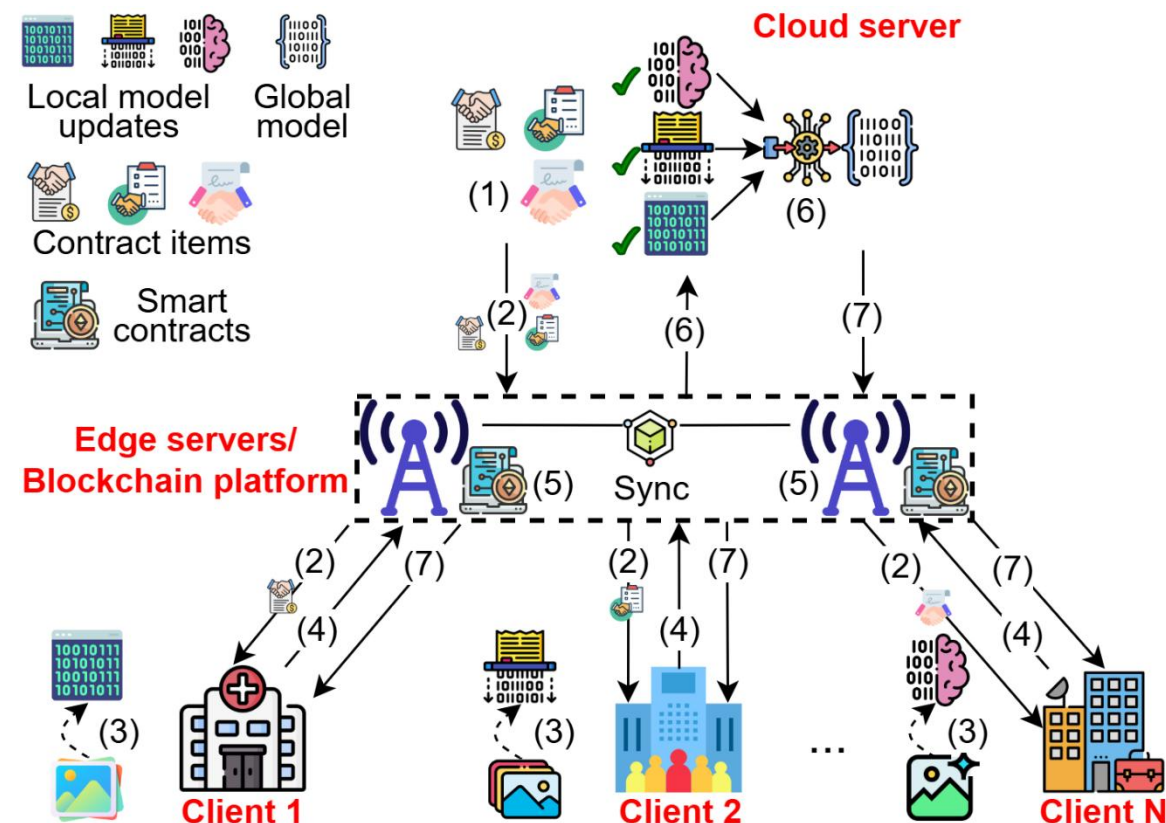
#Research Question: *How can we design an **incentive mechanism** to attract clients with high quality contributions to improve **the FL learning efficiency**, and **mitigate information asymmetry** between clients and server, given the crucial role of **CLPs** in determining FL model performance?*

Proposed Solution – System Architecture

Developing a ***Right Reward Right Time (R3T)*** framework integrating **contract theory** and **blockchain smart contracts**.

System Architecture

- FL clients
 - Have data and choose the time to join the training process of FL global model
- Blockchain platform
 - Validate/Record client information (e.g., ID, contributions)
 - Coordinate the incentivization process
- Cloud server
 - Aggregate the FL global model
 - Make payment to clients



R3T system with learning workflow comprising of (1) Prepare contracts, (2) sign contracts & download global model, (3) local training, (4) upload updates, (5) validate updates, (6) receive & aggregate model updates, and (7) settlement.

Proposed Solution – Problem Statement

- **Set of clients:** $N = \{1, \dots, n, \dots N\}$
- **Set of client types:** $K = \{1, \dots, k, \dots K\}$
- **Set of client's capability:** $\theta = \{\theta_1, \dots, \theta_k, \dots, \theta_K\}$
- **Set of edge servers:** $L = \{l_1, \dots, l_i, \dots, l_L\}$
- **Time frame:** $T = \{1, \dots, t, \dots, T\}$
- **Set of contract items:** $\phi = \{\phi_k\}_{k \in K} = \{(e_k, t_k, R_k)\}_{k \in K}$
- The reward R_k for contributing e_k at t_k is calculated as:

$$R_k = r_k + B_k, \quad (1)$$

$$B_k = \theta_k h(t_k) e_K, \quad (2)$$

where r_k is the base salary, B_k is the bonus, and the *CLP-aware* bonus unit function $h(t_k)$ is defined as follows:

$$h(t_k) = \begin{cases} 1 + \frac{\alpha}{\ln(2t_K)}, & \text{if } t_k \in CLPs \\ 1, & \text{if } t_k \notin CLPs \end{cases}. \quad (3)$$

- Rewrite the contract item: $\{\phi_k\}_{k \in K} = \{(e_k, t_k, r_k)\}_{k \in K}$

Proposed Solution – Utility Model

▪ **Client n utility:**

$$U_n(e_k, t_k, r_k) = \underbrace{R_k}_{\text{Gain}} - \underbrace{\left(\frac{1}{2}\delta e_k^2 + \beta r_k\right)}_{\text{Total contribution costs}} = \theta_k h(t_k) e_k - \frac{1}{2}\delta e_k^2 - (\beta - 1)r_k. \quad (4)$$

▪ **Edge server l utility:**

$$U_l = \underbrace{R_l}_{\text{Gain for data mining}} - \underbrace{v(f_l)}_{\text{Data mining cost function}}, \quad (5)$$

where f_l is the edge server's computing resource.

▪ **Cloud utility:**

$$U_c(e, t, r) = \lambda \underbrace{\sum_{k=1}^K g(h(t_k) e_k)}_{\text{Learning performance gain}} - \underbrace{\sum_{k=1}^K R_k - \sum_{l=1}^L R_l}_{\text{Total rewards paid to clients and edge servers}}, \quad (6)$$

where λ is the cloud server's preference parameter, $g(\cdot)$ is the concave function of performance evaluation.

Proposed Solution – Problem Formulation

- **Game:**

- **Players:** $|S_n| \in N$ clients per training round.
- **Strategies:** $n \in S_n$ chooses corresponding ϕ_k .
- **Objectives:** Each type- k aims to **maximize** its utility expressed in (4).

- Given t_k and r_k , we can get the optimal choice of e_k by setting the first-derivative of (4) as follows:

$$e_k^* = \theta_k h(t_k). \quad (8)$$

- **Global Objective:** Utility **maximization** in cloud server

$$\max_{(t,r)} U_c(e, t, r), \quad (7a)$$

Constraints:

1. Individual Rationality (IR): $U_n(t_k, r_k) \geq 0$, (7b)
2. Incentive Compatibility (IC): $U_n(t_k, r_k) \geq U_n(t_{k'}, r_{k'}), \forall k, k' \in K$. (7c)

Proposed Solution – R3T Incentivization

▪ Under Complete Information Case (CIC):

Lemma 1 (Monotonicity in CLPs). *The higher-type client should receive a higher reward R when they join the training process early and contribute more effort. Under the IC constraints, if $\theta_k > \theta_{k'}, \forall k, k' \in \mathcal{K}$, the reward, effort, and joining time satisfy the following inequalities:*

$$R_k > R_{k'}, r_k \geq r_{k'}, B_k > B_{k'}, e_k > e_{k'}, t_k \leq t_{k'}. \quad (15)$$

Lemma 2 (Client utility condition). *For any feasible contract item (e_k, t_k, r_k) , the type- k client's utility satisfies*

$$\mathcal{U}_n(e_1, t_1, r_1) < \dots < \mathcal{U}_n(e_k, t_k, r_k) < \dots < \mathcal{U}_n(e_K, t_K, r_K)$$

Theorem 1. *All optimal contract items satisfy the condition $\frac{1}{2}\theta_k^2 h^2(t_k) - (\beta - 1)r_k = 0, \forall k \in \mathcal{K}$ under the complete information case, that is the client's utility is zero.*

▪ Under InComplete Information Case (IIC):

Lemma 5 (Monotonicity in non-CLPs). *In the non-CLPs, R3T still ensures incentive fairness by enabling a higher-type client to receive a higher reward. Thus, to satisfy the IC constraints in (7c), if $\theta_k > \theta_{k'}, \forall k, k' \in \mathcal{K}$, then the reward, salary, and effort must adhere to the following inequalities:*

$$R_k > R_{k'}, r_k = r_{k'}, e_k > e_{k'}.$$

Theorem 2. *For any type- k client, the contract salary satisfies*

$$r_k = \sum_{i=1}^k \frac{\theta_i^2 [h^2(t_i) - h^2(t_{i-1})]}{2(\beta - 1)}.$$

Numerical Results – Simulation Setup

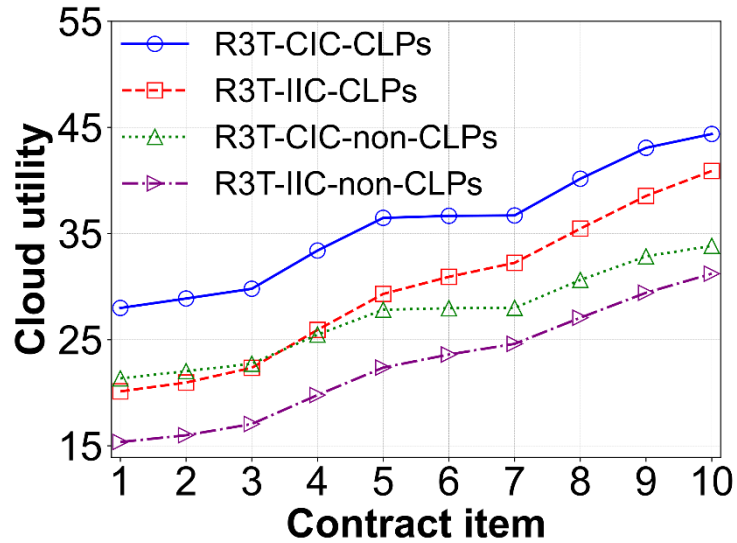
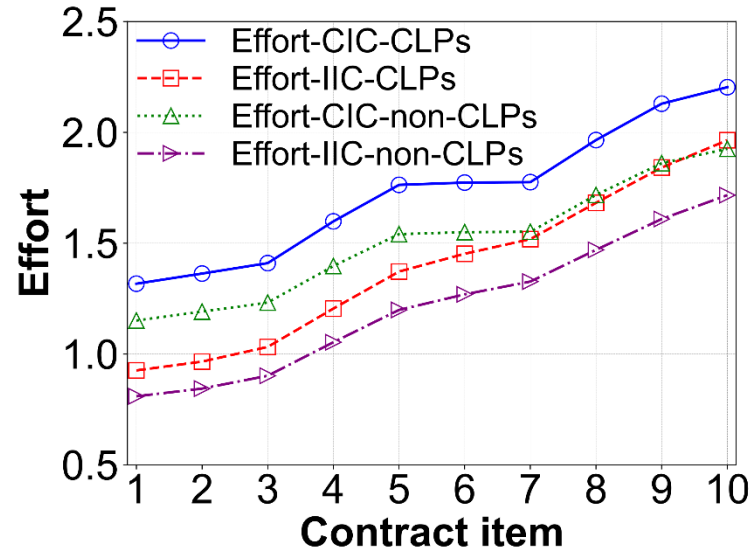
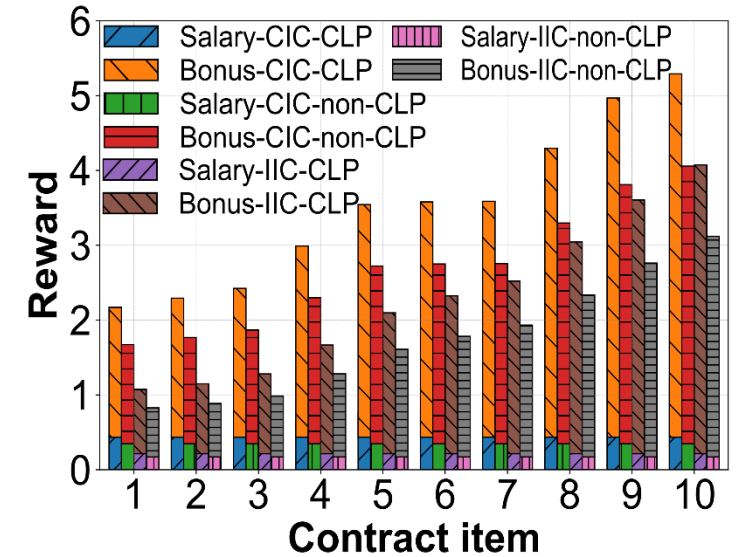
Settings:

- The number of training rounds: $T = 25$
- CLPs: from the training round 1 to training round 10
 - $|S^{t+1}| \leftarrow \min(2|S^t|, N)$
- Non-CLPs: from the training round 11 to training round T
 - $|S^{t+1}| \leftarrow \max\left(\frac{1}{2}|S^t|, \frac{1}{2}N\right)$
- The number of clients: $N = 15$
- The number of contract items: $K = 10$

Benchmarks for comparisons:

- **Linear Pricing:** the cloud determines a unit price P for the client's effort e , where $P = 2.4$.
- **Conventional contract theory without considering the time factor (CTWT):** temporal property is not considered, meaning t is excluded from contract items. **All time slots are treated equally.**
 - Complete Information Cases (CICs).
 - Incomplete Information Case (IICs).

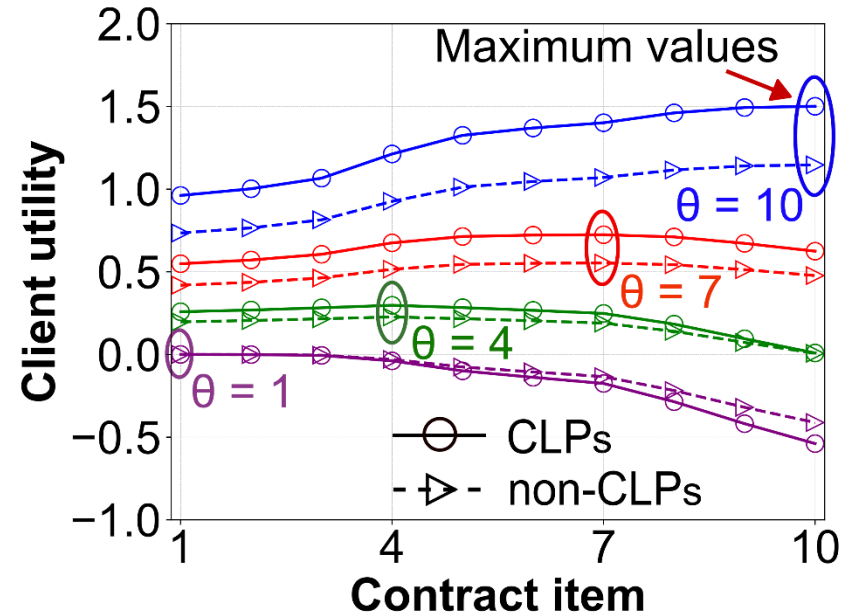
Numerical Results – Key Findings – Feasibility [1/2]



➤ **Obs. 1:** The cloud attains higher utility in both *CLPs* and *non-CLPs* under *CIC* compared to *IIC*.
[Theorem 1]

➤ **Obs. 2:** Higher-type (higher contract item-type) clients contribute more efforts and, consequently, receive higher rewards. [Lemma 1&5]

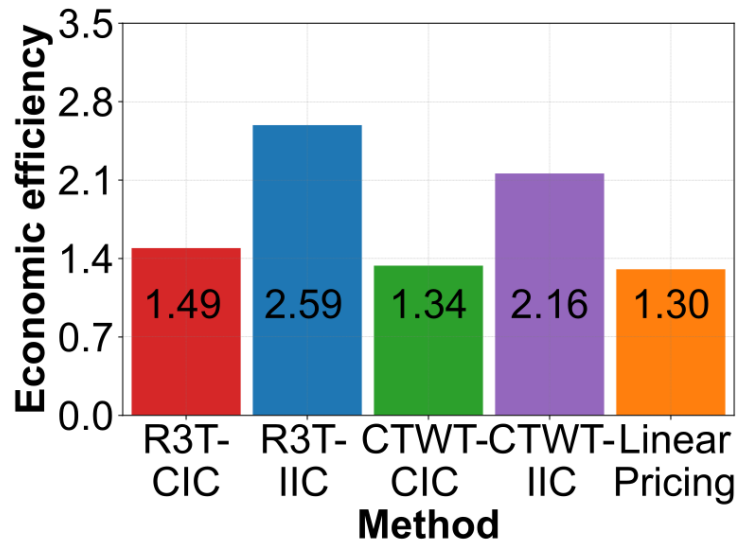
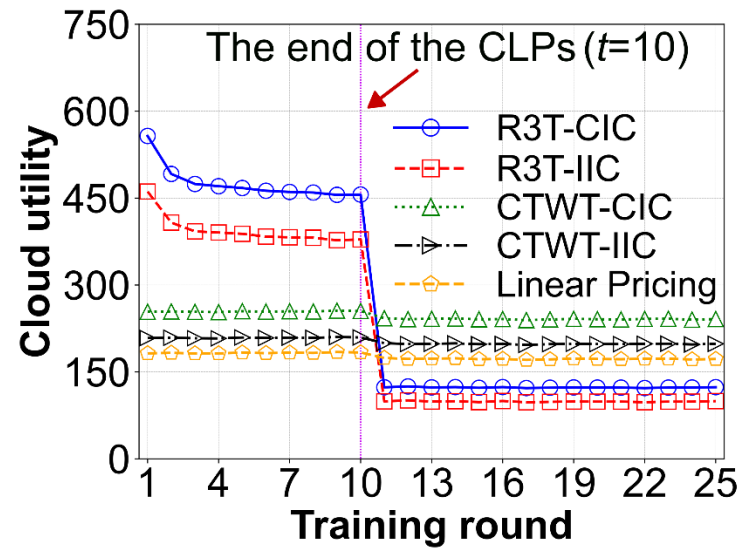
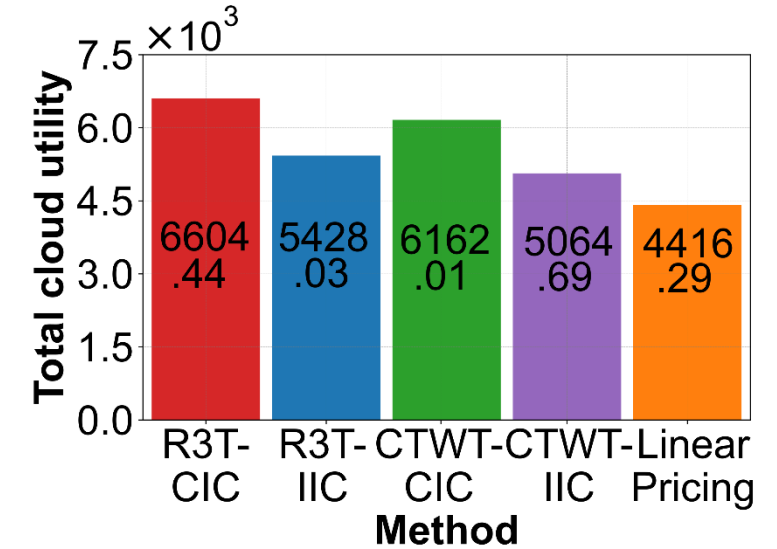
Numerical Results – Key Findings – Feasibility [2/2]



- **Obs. 3:** Four client types select the same contract item $\Rightarrow U_1 < U_4 < U_7 < U_{10}$. [Lemma 1&2]

*This contract design ensures that clients' types are **automatically disclosed** to the cloud server upon contract selection, effectively **addressing information asymmetry** between clients and the cloud server.*

Numerical Results – Key Findings – Efficiency



* *Economic efficiency* reflects the effectiveness of the cloud in calculating and distributing the total amount of reward to clients based on their efforts and joining times.

- **Obs. 1:** R3T spends approximately **20%** and **99%** less on the amount of reward compared with **CTWT** & **Linear Pricing**, respectively.
- **Obs. 2:** R3T achieves total cloud utility up to 7.2% and 22.9% higher compared to **CTWT-IIC** and **Linear Pricing**, respectively.

Pay less, Get even more! Highlighting the importance of providing the right reward at the right time.

Conclusion and Future Work

- This paper leverages CLPs, contract theory and blockchain smart contracts to propose a *CLP-aware incentive mechanism* for FL, named Right Reward Right Time (**R3T**).
- **R3T** characterizes the cloud utility as a function of client's time and system capabilities, effort, joining time, and reward.
- We proposed an optimal contract design that incorporates the temporal factor and then derived the optimal incentive strategies for the cloud server.
- **R3T** incentivizes early participation and efforts while maximizing cloud utility, even under information asymmetry. Compared with other benchmarks, it achieves:
 - Up to **99%** reduction in total rewards distributed.
 - Up to **22.9%** improvement in the cloud utility.
- **Future work:**
 - Examine R3T under different budget constraints and R3T scalability.
 - Build R3T Proof of Concept to show the feasibility and effectiveness in practical setups.

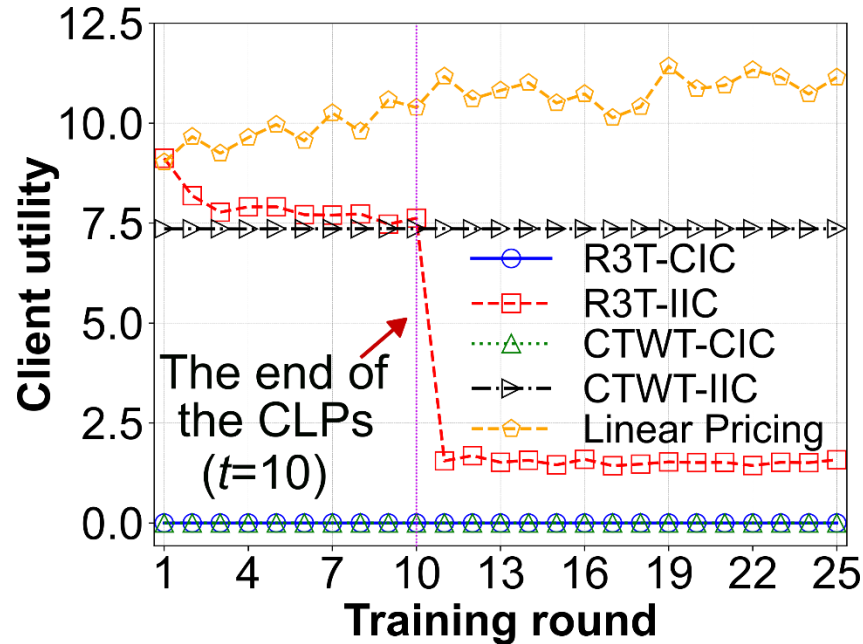
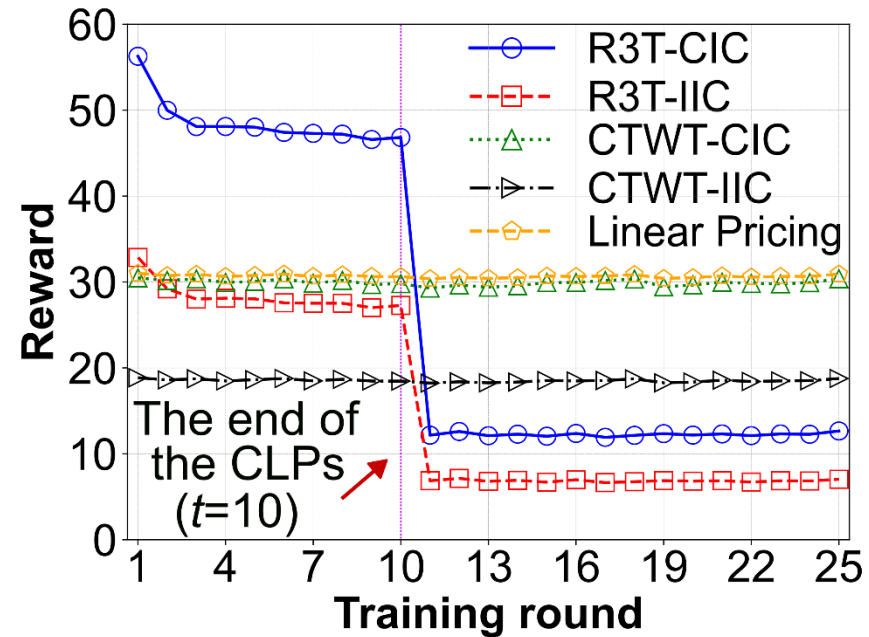


Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

Thank You

Thanh Linh Nguyen
Email: tnguyen3@tcd.ie

Numerical Results – Key Findings – Efficiency



- **Obs. 1:** Clients' utilities in **R3T** decrease after CLPs.
- **Obs. 2:** **CTWT**'s client utilities are stable over T rounds. Especially, client utilities are 0 in **CTWT-CIC**. **[Theorem 1]**
- **Obs. 3:** **Linear Pricing** exhibits the highest client utility.

Appendix: R3T article

The reward structure of R3T is established as follows:

$$R_k = r_k + B_k,$$
$$B_k = \theta_k h(t_k) e_k,$$

where R_k is the corresponding reward for type- k client paid by the cloud server for finishing the training round,
 r_k is the salary,
 B_k is the bonus,
 $h(t_k)$ is a CLP-aware bonus unit function (bonus-time relationship function) [1].

$$h(t_k) = \begin{cases} 1 + \frac{\vartheta}{\ln(2t_k)}, & \text{if } t_k \in CLPs \\ 1 & , \text{if } t_k \notin CLPs \end{cases}$$

[1]. Ahmet Melih Selcuk and Zeynep Muge Avsar, "Dynamic pricing in airline revenue management," *Journal of Mathematical Analysis and Applications*, vol. 478, no. 2, pp. 1191–1217, 2019.

Appendix: R3T article

- The utility of a type- k client n will be the difference in the reward R_k they receive for effort contributions and the total contribution costs

$$U_n(e_k, t_k, r_k) = R_k - \left(\frac{1}{2} \delta e_k^2 + \beta r_k \right),$$

where
 $\underline{e}$ $\frac{1}{2} \delta e_k^2$ is the quadratic cost function with respect to client's effort,
 $\delta > 0$ is the unit effort cost,
 βr_k ($\beta > 0$) represents the time cost associated with joining time t_k ,
 $h(t_k)$ is a CLP-aware bonus unit function (bonus-time relationship function) [1].

- The utility of the cloud server is as follows

$$U_c(\mathbf{e}, \mathbf{t}, \mathbf{r}) = \lambda \sum_{k \in K} g(h(t_k) e_k) - \sum_{k \in K} R_k - \sum_{l \in L} R_l,$$

where
 $\underline{e}$ λ is a parameter adjusted by the cloud server to show its concerns for FL model performance,
 $\delta > 0$ is the unit effort cost,
 $g(\cdot)$ is defined as a concave function.

Appendix: R3T article

- We aim to maximize the cloud utility while ensuring IR, IC, and BF constraints:

$$\begin{aligned} & \max_{(\mathbf{e}, \mathbf{t}, \mathbf{r})} U_c(\mathbf{e}, \mathbf{t}, \mathbf{r}), \\ & \text{s.t., } U_n(e_k, t_k, r_k) \geq 0, & [\text{IR}] \\ & \quad U_n(e_k, t_k, r_k) \geq U_n(e_{k'}, t_{k'}, r_{k'}), & [\text{IC}] \\ & \quad \sum_{k=1}^K R_k \leq P, \forall k, k' \in \mathbf{K}, & [\text{BF}] \end{aligned}$$

wher P is the total cloud budget for a learning-based task.

e

Once t_k and r_k are given, we can obtain e_k by setting the first-order derivative of the U_n , then we have:

$$e_k^* = \frac{\theta_k h(t_k)}{\delta}$$

Appendix: R3T article

R3T under Complete Information Case (CIC)

Theorem 1: All optimal contract items satisfy the condition $\frac{\theta_k^2 h^2(t_k)}{2\delta} - (\beta - 1)r_k = 0, \forall k \in K$, under CIC, that is the client's utility is zero.

By replacing $r_k = \frac{\theta_k h^2(t_k)}{2\delta(\beta-1)}$, we have the following optimization problem:

$$\begin{aligned} \max_t \quad & \sum_{k=1}^K \left[\lambda g\left(\frac{\theta_k h^2(t_k)}{\delta}\right) - \frac{\theta_k^2 h^2(t_k)}{2\delta(\beta-1)} - \frac{\theta_k^2 h^2(t_k)}{\delta} \right], \\ \text{s.t.} \quad & \sum_{k=1}^K \left[\frac{\theta_k^2 h^2(t_k)}{2\delta(\beta-1)} - \frac{\theta_k^2 h^2(t_k)}{\delta} \right] \leq P, \forall t_k \in T, \forall k \in K. \end{aligned}$$

→ Using the exhaustive search algorithm to find optimal values of the t_k , r_k , R_k , and e_k .

Appendix: R3T article

R3T under Incomplete Information Case (IIC)

In CLPs

Theorem 2: For any type- k client, the contract salary satisfies:

$$r_k = \sum_{i=1}^k \frac{\theta_i [h^2(t_i) - h^2(t_{i-1})]}{2\delta(\beta - 1)}.$$

By using $r_k = \frac{\theta_k h^2(t_k)}{2\delta(\beta-1)}$, we have the following optimization problem:

$$\begin{aligned} \max_t \sum_{k=1}^K & \left[\lambda g \left(\frac{\theta_k h^2(t_k)}{\delta} \right) - \sum_{i=1}^k \frac{\theta_i [h^2(t_i) - h^2(t_{i-1})]}{2\delta(\beta-1)} - \frac{\theta_k^2 h^2(t_k)}{\delta} \right], \\ \text{s.t. } & \underbrace{R_k > R'_k, r_k \geq r'_k, B_k > B'_k, e_k > e'_k, t_k \leq t'_k}_{\text{Monotonicity}}, \forall t_k \in T, \forall k \in K. \end{aligned}$$

Monotonicity

*** Note that at $i = 0$, we have $h(t_0) = 0$ because t_0 is the time before the training process starts.

In non-CLPs

$$\begin{aligned} \max_t \sum_{k=1}^K & \left[\lambda g \left(\frac{\theta_k h^2(t_k)}{\delta} \right) - \sum_{i=1}^k \frac{\theta_i [h^2(t_i) - h^2(t_{i-1})]}{2\delta(\beta-1)} - \frac{\theta_k^2 h^2(t_k)}{\delta} \right], \\ \text{s.t. } & R_k > R'_k, r_k = r'_k, B_k > B'_k, e_k > e'_k, \forall t_k \in T, \forall k \in K. \end{aligned}$$

$$r_k = r_{k'} = r_1 = \frac{\theta_1^2 h^2(t_1)}{2\delta(\beta - 1)}.$$

Remark 1. In non-CLPs, the impact of the client's contribution is less significant compared to CLPs. As Lemma 5 outlines, although all clients receive a fixed salary regardless of their types and do not receive weighted bonuses (i.e., $h(t_k) = 1$) for specific joining times, R3T ensures the fairness of their total rewards relative to their corresponding efforts.

Arguments about FML privacy in paper

- ❖ These critiques (i.e., Trust dependency, Architectural vulnerabilities, Limited added privacy with trustworthy server) are valid but reflect limitations in *current implementations* of FL, not inherent flaws in the paradigm itself. FL's privacy benefits are achievable with proper safeguards and evolving techniques.
- ❖ *Core advantages of FL:* **Data localization**
 - Data minimization
 - Compliance: GDPR, HIPAA
 - Attack surface reduction
 - ➔ While the paper argues that centralized training with ephemeral data could match FL privacy, this assumes servers delete data immediately. FL inherently enforces data localization, making accidental or intentional misuse harder.
- ❖ *Address Trust and Security Challenges:*
 - Reducing trust on the server: using verifiable computation (TEEs), blockchain-based FL

1. Lawful Basis for Processing (Article 6)

- **Requirement:** The cloud server must establish a lawful basis for collecting and processing client data. Common bases include:
 - **Explicit consent** from the client.
 - **Contractual necessity** (e.g., data processing required to fulfill a service).
 - **Legitimate interests** (e.g., improving system performance, fraud prevention).
 - **Action:** Clearly inform clients of the purpose of data collection and obtain unambiguous consent where required.
-

2. Data Minimization (Article 5)

- **Requirement:** Only collect data that is **necessary, relevant, and limited** to the stated purpose.
 - **Action:** Avoid collecting excessive or irrelevant data (e.g., avoid gathering sensitive personal data unless absolutely required).
-

3. Transparency and Accountability (Articles 12–14)

- **Requirement:** Provide clients with clear, accessible information about:
 - What data is collected.
 - How it will be used.
 - Who it will be shared with.
 - Data retention periods.
- **Action:** Publish a GDPR-compliant privacy policy and ensure clients can easily access it.

4. Why Potential Games Are Better Suited for FL

- Unlike contract theory, Stackelberg games, and cooperative games, **potential games** are well-suited to model the **decentralized and competitive dynamics** of FL because:
 - They do not require asymmetric information or complex incentive mechanisms.
 - They do not assume a leader-follower hierarchy or the formation of coalitions.
 - They naturally capture the interplay between individual client objectives and the global goal of improving the model.
 - They guarantee convergence to a stable state (pure Nash equilibrium), which is critical for ensuring the stability and efficiency of FL systems.