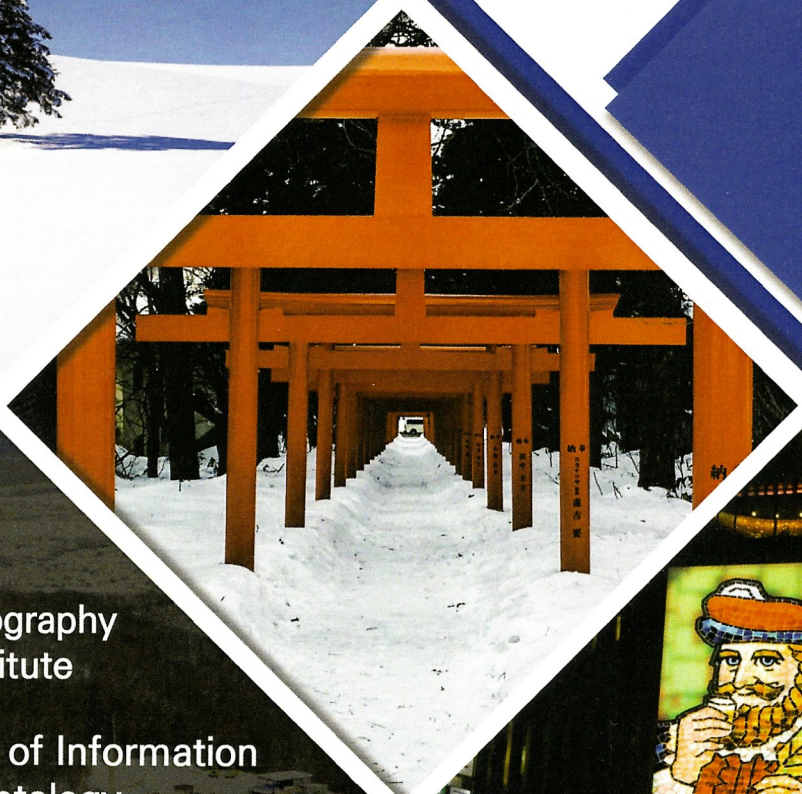
MobiSec 2024

The 8th International Conference on Mobile Internet Security

Hotel emisia,
Sapporo, Japan
December 17 – 19, 2024



| Organizer :
- KIISC Research Group on 6G Security
- Kookmin University Cryptography & Information Security Institute

| Host : Korea Institute of Information Security & Cryptology

| Technically Sponsored by : IEICE ICSS, JSAI-SIG-SEC

| In Cooperation with:
- IEICE ISEC
- Kookmin University BK21 Four Institute of Information Security Education for Secure Hyperconnected Society

Day 3 Room 1

12/19 (Thu.)

Time / Session	Presentation Paper	Article No.
Session 8A 09:00 ~ 10:15 AI-Driven Security II (Offline)	**SCU-CGAN: Enhancing Fire Detection through Synthetic Fire Image Generation and Dataset Augmentation** Ju-Young Kim, Ji-Hong Park (Gyeongsang National University) Gun-Woo Kim (Gyeongsang National University)	68
	Clock Glitch-based Fault Attacks on Convolutional Neural Networks Seongwoo Hong, Hyoju Kang, Seungyeol Lee, Jaecheol Ha (Hoseo University)	69
	Federated Random Forests for Privacy-Preserving Intrusion Detection in IoT Networks Md. Parvezur Rahman Mahin, Farhana Anwar Tisha (East West University) Effat Ara (American International University of Bangladesh) Raihan Ul Islam (Associate Professor, East West University) Karl Andersson (Lulea University of Technology) Mohammad Shahadat Hossain (University of Chittagong)	70
	Fair Incentive Distribution Mechanism in Hierarchical Federated Learning Siwan Noh, Ju-Hyun Jeon, Kyung-Hyune Rhee (Pukyong National University)	71
BREAK TIME (15 Min)		
Session 9A 10:30 ~ 12:00 Cyptography and Authtenticaiton (Offline)	**Bidirectional Proxy Re-Encryption based on Isogenies** Jiawei Chen, Hyungrok Jo, Shingo Sato, Junji Shikata (Yokohama National University)	80
	PRNG-Oriented Side-Channel Security Evaluation for TI-AES Yusaku Harada, Maki Tsukahara, Daiki Miyahara, Yang Li, Kazuo Sakiyama (The University of Electro-Communications) Yuko Hara (Tokyo Institute of Technology)	81
	A Blockchain-Based Approach for Secure Email Encryption with Variable ECC Key Lengths Selection Md. Biplob Hossain, Maya Rahayu, Yuta Kodera, Yasuyuki Nogami, Samsul Huda (Okayama University) Md. Arshad Ali (Faculty of CSE, Hajee Mohammad Danesh Science and Technology University)	82
	Analysis of Numerous Security Algorithm Performance with Data Encryption on Edge Device Sangmyung Lee, Ehan Sohn, Soeun Kim (Seoul National University of Science and Technology) Sunggon Kim (Seoul National University of Science and Technology)	83
	Lightweight IoT Data Encryption using Time Parameter based Ascon Algorithm Kunlin Tsai, Ju-Wei Zhu, Deng-Yao Yao, Guo-Wei Wang, Fang-Yie Leu (TungHai University)	84

Fair Incentive Distribution Mechanism in Hierarchical Federated Learning*

Siwan Noh, Ju-Hyun Jeon, and Kyunh-Hyune Rhee†

Pukyong National University, Busan, 48513, South Korea
{nosiwan, jhjeon, khrhee}@pukyong.ac.kr

Abstract

The integration of artificial intelligence (AI) in healthcare, powered by Internet of Medical Things (IoMT) data, offers significant potential for personalized and efficient patient care. Hierarchical federated learning (HFL) is a promising approach for healthcare applications, combining cross-silo and cross-device federated learning. This architecture allows hospitals to train local models using patient data, while sharing anonymized parameters with other hospitals to improve diagnosis. However, existing studies on incentive mechanisms in HFL often focus on determining optimal incentive values but neglect the integration of these incentives into the reward stage. Moreover, the two-layer architecture of HFL introduces challenges related to disparities in patient volume and diversity across hospitals. In this paper, we propose a fair incentive distribution mechanism for hierarchical systems using blockchain state channels. We ensure equal incentive budget contributions from all organizations, preventing free-riders in the HFL system with a channel factory solution. Additionally, virtual channels support transactions without intermediaries, minimizing computational costs in the blockchain network.

Keywords: H ierarchical federated learning, Incentive mechanism, State channel

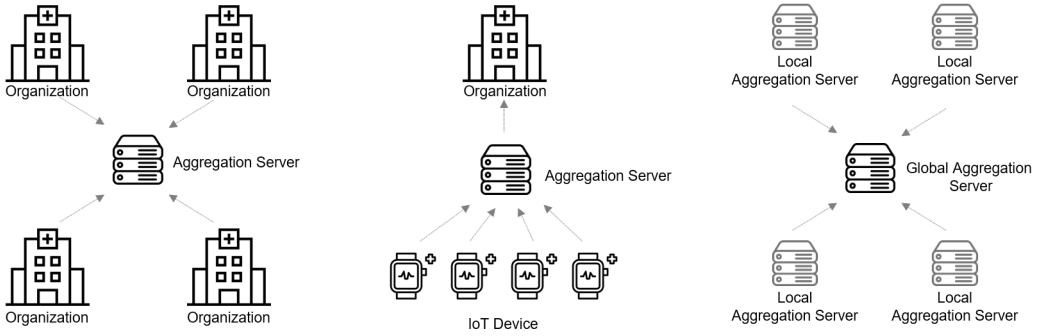
1 Introduction

The integration of Artificial intelligence (AI) into healthcare has the potential to revolutionize the way medical professionals approach patient care, diagnosis, and treatment. With the vast amount of data available in healthcare has captured by the Internet of Medical Things (IoMT), AI can help to analyze and interpret this data to provide personalized and efficient care[1, 2]. IoMT is a network of medical devices, implants, and wearables that collect and transmit patient data. This data can be used to improve healthcare in a variety of ways, such remote patient monitoring, personalized medicine, and clinical research. However, there are also some challenges associated with using IoMT data in healthcare. One of the biggest challenges is privacy. IoMT data is very sensitive, and it is important to protect it from unauthorized access.

Federated learning (FL) addresses privacy concerns in healthcare by enabling collaborative model training without data exchange. Data stays secure on individual devices (e.g., wearables) while the model learns from all participants. According to implementation scenarios, there are three main implementations: cross-silo FL, cross-device FL, and hierarchical FL(HFL)[3]. As shown in Figure 1(a), Cross-silo FL consists of several organizations with large computing power and stable network connections that allow them to keep participating part in training.

*Proceedings of the 8th International Conference on Mobile Internet Security (MobiSec'24), Article No. 71, December 17-19, 2024, Sapporo, Japan.

†Corresponding author



(a) Cross-silo FL (b) Cross-device FL (c) Hierarchical FL

Figure 1: Types of Federated Learning

In contrast, Cross-device FL shown in Figure 1(b) aggregates data from a large number of low-power, heterogeneous devices, sporadic network connectivity issues or a lack of computational power may prevent certain devices from contributing training results. HFL is a model that combines cross-device and cross-silo, where aggregation is performed in two layers as shown in Figure 1(c). HFL is particularly useful in healthcare. For example, hospitals can train local models using patient EHRs(intra-silo aggregations) and then share anonymized parameters with other hospitals(inter-silo aggregations) for improved their diagnosis experiences, all without sharing sensitive data. However, without incentive nobody allow their EHR.

Incentives are crucial in many systems to encourage participation and ensure high-quality contributions. In FL, incentive mechanisms play a crucial role in encouraging data owners to participate in the collaborative training process. By providing appropriate rewards, these mechanisms can motivate data owners to share their valuable data and contribute to the development of shared models. Incentive mechanisms are the cornerstone of encouraging participation in collaborative systems. They operate in two key stages: (1) evaluation and (2) reward.

- **Evaluation Stage**: During this stage, individual contributions are assessed using a predefined method. This ensures fairness and consistency in determining who deserves a reward. Quality-aware mechanisms, for example, take into account the quality of contributions alongside quantity. This motivates participants to prioritize accuracy and effort, ultimately leading to better overall results. Reputation also plays a role in some systems. These reputation-aware systems consider an individual's past behavior, feedback from others, and ratings when making decisions about rewards. This helps identify reliable and valuable contributors, further incentivizing positive behavior within the system.

- **Reward Stage**: Based on the evaluated metrics, rewards are distributed to participants. Monetary compensation is a straightforward approach, offering direct payments for training participation or bonuses for high-quality data contributions. Alternatively, reputation systems can be employed, assigning scores or badges based on contributions, participation levels, and data quality. These scores can then unlock privileges, such as access to exclusive features or priority in future training rounds.

Recently, various incentive mechanisms have been proposed to accurately and fairly measure user contributions in HFL-based systems[4, 5, 6, 7]. These studies proposed utilizing game

theory and contract theory to determine optimal incentives based on user contributions[4, 5, 6], or to provide incentives based on a comprehensive scoring system that aggregates evaluation results throughout the entire learning process[7]. While many studies focused solely on determining the optimal incentive value, they fail to consider the integration of these incentives into the reward stage for users. Furthermore, the two-layer architecture of the HFL system model for IoMT applications, comprising a local layer of individual hospitals and patients and a global layer of a network of hospitals, introduces challenges due to the inherent independence of these hierarchical layers. Each hospital generates local models based on the diversity of patients within its organization and shares its clinical experience with all hospitals in the system through federated learning in the global layer. While federated learning can benefit from a large and diverse patient pool, disparities in patient volume and diversity across hospitals can lead to vulnerabilities like free-riding attacks, where hospitals with minimal contributions reap the same benefits.

In this paper, we propose a fair incentive distribution mechanism in a hierarchical system based on blockchain state channels. Our mechanism ensures that all organizations, regardless of size, invest an equal incentive budget in model development to prevent free-riders in the HFL system. This is implemented using a state channel solution known as a channel factory. Additionally, considering the iterative training over multiple rounds, we utilize virtual channels that support transactions without intermediaries in a hierarchical architecture, thereby minimizing computational costs associated with resource consumption in the blockchain network. To summarize, the major contributions of this paper are as follows:

1. We introduce a HFL system leveraging blockchain state channels to ensure fair incentive distribution.

2. Our mechanism mandates equal incentive budget contributions from all organizations, irrespective of their size, to prevent free-riding in the HFL system. This is achieved through the implementation of a channel factory.

3. We utilize virtual channels to support transactions without intermediaries in a hierarchical architecture, minimizing computational costs associated with blockchain resource consumption.

2 Related Work

Existing literature has proposed incentive mechanisms in FL systems to evaluate users' contributions more accurately and to determine fair rewards. However, they also have some limitations that need to be addressed. One of the limitations is that the free-rider problem in HFL environments has not been considered, which can discourage the participation of large superorganizations with diverse and rich participant pools. Another limitation is the absence of a suitable reward distribution and settlement mechanism for complex hierarchical system structures. While blockchain-based approaches provide irreversible settlement records, the complexity of representing hierarchical structures inevitably leads to increased processing costs. Table 1 presents a detailed comparison between our proposed work and the existing incentive mechanisms in federated learning systems.

With insight into the recent trends and research challenges for incentive mechanisms in FL systems and the drawbacks mentioned above of the existing studies, we propose a blockchain-based approach for HFL, combining a channel factory[12] and virtual channels[13] to represent

Table 1: Comparative Summary of Incentive Mechanism

Proposal	Evaluation Metrics	Incentive Mechanism	Reward	FL Model
[8]	Contribution quality	Scoring	Token-based	Cross-silo
[9]	Contribution quality, User repuation	Game theory (Repeated gamses)	N/A	Cross-silo
[10]	Social efficiency, Individual rationality, Budget balance	Social wellfare maximization	N/A	Cross-silo
[11]	Contribution quality, User reputation	Scoring	Token-based	Cross-device
[4]	Client rationality	Game theory (Stackelberg game), Contract theory	N/A	Hierachical
[5]	Contribution quality	Contract theory	N/A	Hierachical
[6]	Training Efficiency	Game theory (Stackelberg game)	N/A	Hierachical
[7]	Contribution quality	Scoring	N/A	Hierachical
Proposed	Flexible	Flexible	Token-based	Hierachical

complex hierarchical structures on a blockchain. This addresses free-riding by settling inter-layer accounts through a top-down tree of state channels, with final settlement at the top level.

3 Proposed System

This section presents an informal description of our mechanism. Our proposed mechanism can be constructed as a blockchain layer on top of existing FL architectures to distribute rewards determined in a FL system where hierarchical participants are involved. The term *hierarchical participants* in this case can encompass large hospitals with multiple branches or subsidiaries, or healthcare organizations under a medical foundation.

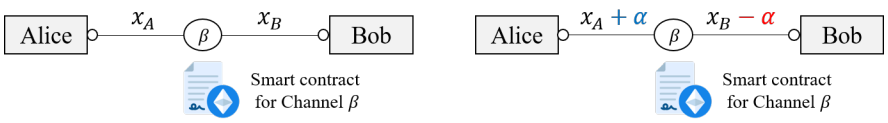
3.1 Preliminary

3.1.1 State Channels

State channels[1] are a scaling solution designed to significantly increase the transaction throughput and reduce costs on blockchain networks. Instead of broadcasting every transaction to the network, state channels allow participants to conduct multiple transactions off-chain. For instance, when two parties, Alice and Bob, engage in a transaction via a channel, as depicted in Figure 2(a), they deposit their fund x into a smart contract on the blockchain. This smart contract acts as an intermediary, holding the funds $x_A + x_B$ until the channel is closed. The initial state of the funds S_0 is represented by

$$[Alice \rightarrow x_A, Bob \rightarrow x_B] \tag{1}$$

[1]Ethereum State Channels, https://ethereum.org/en/developers/docs/scaling/state-channels/



(a) Channel Creation (b) Channel Update

Figure 2: Overview of State Channels

When Bob pays Alice an amount of $\alpha \leq x_B$ as shown in Figure 2(b), the updated state of the funds S_1 can be represented as

$$[Alice \rightarrow x_A + \alpha, Bob \rightarrow x_B - \alpha] \tag{2}$$

A channel update is a transition from the initial state to a new state, defined as

$$S_i = f(S_0, t_i) \tag{3}$$

where, f represents the channel update function, and t denotes a transaction that involves a transfer of funds from an initial state to a new state.

These updates can be performed multiple times until the channel is closed, and throughout this process, the total value of funds never changes. With the exception of the initial state S_0, all intermediate channel states remain off-chain and are shared exclusively between the channel participants. Channel closure, which requires consensus among all participants, results in the submission of the final state to the blockchain. Consequently, despite multiple off-chain state transitions, the blockchain observes only a single state transition upon channel closure. For instance, after five channel updates, the blockchain records the following state transition: $S_5 = f(S_0, t_5)$.

3.1.2 Channel Factory

The channel factory[12], proposed by Burchert et al., is a new layer designed to improve the scalability of Bitcoin micro-payment channels. Unlike traditional one-to-one channels, it allows the creation of multiple sub channels from the funds of multiple groups of parties. One notable feature is that the total value of funds allocated to the channels can be changed without closing the channels. For example, assuming that parties Alice, Bob, Carol, and David create sub-channels through a channel factory, the initial state of the channels S_0 can be represented as

$$[Alice \rightarrow x_A, Bob \rightarrow x_B, Carol \rightarrow x_C, David \rightarrow x_D] \tag{4}$$

The new state of the channel updates the balances of the parties' funds within the channel through Equation 3. The channel factory uses the balance of S_0 to create multiple sub-channels. For example, it can create two sub-channels from S_0,

$$[Alice \rightarrow x_A, Bob \rightarrow x_B], [Carol \rightarrow x_C, David \rightarrow x_D] \tag{5}$$

or it can split the funds to be used in different channels(where, $x'_A + x''_A + x'''_A = x_A$).

$$[Alice \rightarrow x'_A, Bob \rightarrow x_B], [Alce \rightarrow x''_A, Carol \rightarrow x_C], [Aclie \rightarrow x'''_A, David \rightarrow x_D] \tag{6}$$

The new state S_1 of the channel, created in this manner, is composed of the combination of state transitions of multiple sub-channels as described above. Therefore, the state transition of the channel in the channel factory can be represented as

$$S_i = f(S_0, t_{i,1}, ..., t_{i,n}) \tag{7}$$

where, $t_{i,n}$ is a transaction that includes the transfer of funds in sub-channel n.

3.1.3 Virtual Channels

The concept of state channels can be extended to a channel network[14], which enable routing between users who cannot be directly connected by employing intermediary hubs. However, this method required the involvement of intermediaries during communication, leading to increased costs. The Perun[13] addresses this issue by creating a single virtual channel across multiple channel networks. This approach eliminates the need for intermediary participation except during the creation and settlement of the channel. For instance, as illustrated in Figure 3, when Ingrid acts as an intermediary between Alice and Bob, they each establish channels β_A and β_B with Ingrid, respectively. In this scenario, Alice and Bob deposit their funds, y_A and z_B, into their respective channels with Ingrid, who in turn deposits her funds, y_I and z_I, into the channels with both parties. In [14], it was required that when Alice wanted to make a payment to Bob, the state of channel β_A had to be updated first, followed by Ingrid's confirmation, and then the state of channel β_B had to be updated sequentially. However, in a virtual channel, payments were achieved without the need for Ingrid's confirmation as follows.

To open the virtual channel γ, as illustrated in Figure 3(a), Alice, Bob, and Ingrid temporarily remove a certain amount of assets from their deposited funds in their respective channels to be used in the virtual channel. This removal is a logical operation that occurs only among the participants opening the virtual channel and does not actually take place on the blockchain. Subsequently, the balance of funds in the opened virtual channel is given by $[Alice \rightarrow x_A, Bob \rightarrow x_B]$. However, the removed value should not make the funds in the existing channel negative(i.e., $x_A \leq min(y_A, z_I)$ and $x_B \leq min(y_I, z_B)$). The virtual channel created in this manner can be used in the same way as an existing state channel, and Ingrid's participation is not required for channel updates. Once the balance update is complete, if the final balance of the channel is given by $[Alice \rightarrow x'_A, Bob \rightarrow x'_B]$, the closure of the virtual channel updates the states of channels β_A and β_B as shown in Figure 3(b). After the update, the balances of each channel are given by $[Alice \rightarrow y_A - x_A + x'_A, Ingrid \rightarrow y_I - x_B + x'_B]$ and $[Ingrid \rightarrow z_I - x_A + x'_A, Bob \rightarrow z_B - x_B + x'_B]$, and the transfer of funds between Alice and Bob is facilitated through Ingrid.

3.2 Fair Incentive Distribution Mechanism

The top-level hospitals H_i participating in the system aim to develop an AI model for medical diagnosis in collaboration with other hospitals. In this process, H_i train the local model using medical data held by sub-hospitals $H_{i,j}$ within their organizations. The local models generated in this way are collected among the H_i to create a global model, and this process is repeated until the model achieves the desired performance. In this paper, we classify hierarchical hospital organizations into top-level hospitals and their sub-hospitals. For example, hospital H_1 includes an cancer hospital $H_{1,1}$ and a cardiovascular hospital $H_{1,2}$ as shown in Figure 4. The cancer hospital further comprises a liver cancer center $H_{1,1,1}$ and a colorectal cancer center $H_{1,1,2}$. This organization is structured with Hospital H_1 as the top-level hospital and the other hospitals as sub-hospitals.

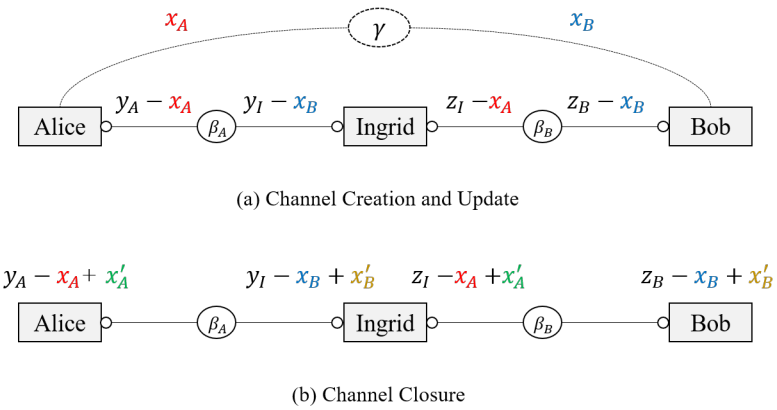


Figure 3: Overview of Virtual Channels

Before we go into the details of these stages, let us briefly describe the overall workflow of the our system. *1) Channel Establishment*: The process begins by creating channels between the hierarchical levels of hospitals. The top-level hospital H_i and its sub-hospitals $H_{i,j}$ create a channel factory. H_i deposits an amount sufficient to incentivize the sub-hospitals, while the sub-hospitals deposit only the minimum amount required as a security deposit to create the channel. Once the channel factory is created, sub-channels are created using the total funds of H_i divided by the number of $H_{i,j}$. For example, as shown in Figure 4, if there are two top-level hospitals and three sub-hospitals, the funds for each sub-channel are calculated by dividing the total funds of the top-level hospitals by the number of child nodes. The sub-hospitals then create state channels with their child nodes, depositing an amount equal to the amount allocated to each sub-channel. This ultimately forms a sub-tree with $H_{i,j}$ as the root node and the patients as the leaf nodes. If, there are no further sub-hospital under the child nodes of H_i, state channels are created only between $H_{i,j}$ and the patients(e.g. $H_{1,2}$ in Figure 4). However, if there are additional sub-organization $H_{i,j,k}$ under $H_{i,j}$, state channels are created between each organization. Using these state channels, virtual channels are then established to facilitate direct transactions between the organizations, which are the sub-hospital $H_{i,j}$, and the leaf nodes, which are the patients of $H_{i,j}$. *2) Channel Update*: During the learning process, the parent nodes in the tree evaluate the local models submitted by their child nodes and update the balance of the channel's funds based on the evaluation results. If the incentives to be paid to a specific sub-tree exceed the funds available in the channel, the top-level hospitals can adjust the fund allocation among the sub-channels to address this issue. 3) *Settlement*: After all learning process is completed, all created channels retain an off-chain state that has not been submitted to the blockchain. Settlement begins with the state of the channel factory being submitted to the blockchain. During this process, each sub-hospital receives the incentive amounts to be paid to their child nodes from the root node. Ultimately, these amounts are settled through the state channels with the leaf nodes, which are the patients, and are paid to each patient.

Let us now define the procedures for these channel operations. We assume that, as shown in Figure 4, the participating organizations in the system form a forest consisting of multiple disjoint trees, each with a top-level hospital as the root node. In a forest with multiple disjoint trees, each tree is identified by its root node. Let T_i be the tree with root node H_i, the

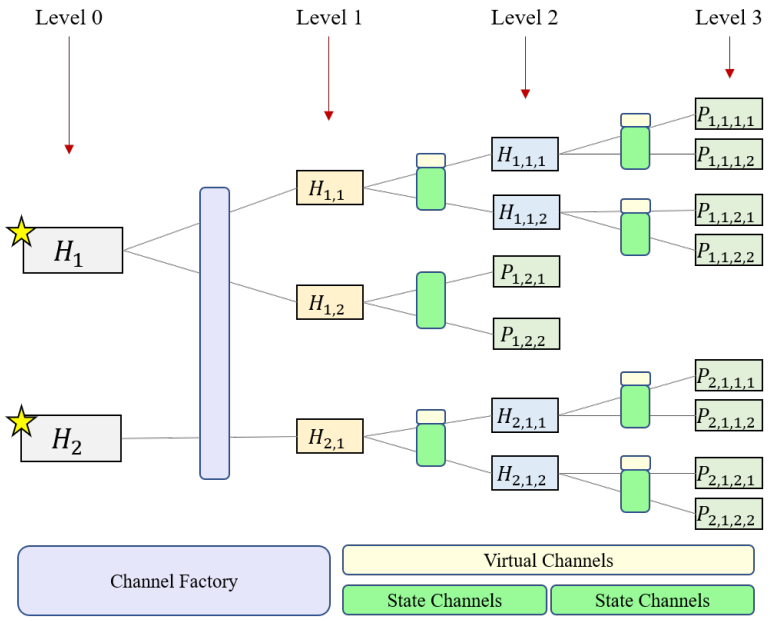


Figure 4: Proposed System Architecture

hierarchical identifier for a node in T_i can be denoted as $H_{i,j,k}$, where i ranges over the root nodes, j ranges over the children of the i-th root node(i.e., $1 \leq j \leq degree(H_i)$), and k ranges over the children of the j-th node(i.e., $1 \leq k \leq degree(H_{i,j})$).

1. *Channel Establishment*: The root node hospitals H_i create a channel factory with sub-hospitals $H_{i,j}$, and the initial state S_0 of the channel factory is

$$S_0 : [H_1 \rightarrow x_1, ..., H_i \rightarrow x_i, H_{1,1} \rightarrow x_{1,1}, ..., H_{i,j} \rightarrow x_{i,j}] \tag{8}$$

where, x is their funds locked in the channel. Channel participants then create sub-channels involving the root node and its child nodes from the channel factory. The number of sub-channels is equal to the sum of the degrees of all root nodes. The amount of funds allocated to the root node for each sub-channel is adjusted flexibly according to the size of the sub trees. The state $S_0^{i,j}$ of the sub-channels created in this manner is represented by

$$S_0^{i,j} : [H_1 \rightarrow x'_{i,j}, ..., H_i \rightarrow x'_{i,j}, H_{i,j} \rightarrow x_{i,j}] \tag{9}$$

where, x' is the portion of the root nodes H_i's funds allocated to the sub-channel for the child node $H_{i,j}$(i.e., $\sum_{i=1}^{N} \sum_{j-1}^{deg(i)} x'_{i,j} = x_i$, where, N is the number of level 0 nodes H_i, and $deg(i)$ is a degree of nodes H_i).

After the sub-channel creation is completed, sub-hospitals $H_{i,j}$ create a state channel $\beta_{i,j}$ and an initial state $S_0^{\beta_{i,j}}$ with their child nodes as

$$S_0^{\beta_{i,j}} : [H_{i,j} \rightarrow x''_{i,j}, H_{i,j,1} \rightarrow x_{i,j,1}, ..., H_{i,j,k} \rightarrow x_{i,j,k}] \tag{10}$$

This step is repeated until the child nodes of sub-hospitals become the leaf nodes of the tree(i.e., patient P). However, $H_{i,j,k}$ deposits $x'''_{i,j,k}$ as the fund for creating the channel

$\beta_{i,j,k}$ with the patient, where, $\sum_{k=1}^{deg(i,j)} x'''_{i,j,k} = x''_{i,j}$. After the creation of all state channels is completed, all level 1 sub hospitals $H_{i,j}$, except those whose child nodes are leaf nodes, create virtual channels γ that enable direct transactions with all leaf nodes included in the sub-tree. The initial state of the channel $\gamma_{i,j,k,l}$ between patient $P_{i,j,k,l}$ and sub-hospital $H_{i,j}$ is represented by

$$S_0^{\gamma_{i,j,k,l}} : [H_{i,j} \rightarrow y_{i,j,k,l}, P_{i,j,k,l} \rightarrow z_{i,j,k,l}] \tag{11}$$

where, $y_{i,j,k,l} \leq min(x''_{i,j}, x'''_{i,j,k})$, $z_{i,j,k,l} \leq min(x_{i,j,k}, x_{i,j,k,l})$ and $\sum_{l=1}^{deg(i,j,k)} y_{i,j,k,l} \leq x'''_{i,j,k}$.

2. *Channel Update*: In this step, a sub-hospitals evaluates users' contributions during the training process. Based on their contributions, incentives are calculated, and the channel state $S_0^{\gamma_{i,j,k,l}}$ is updated to redistribute the funds locked in the channel to reward the users accordingly.

   After the patient submits their local model, the corresponding sub-hospital evaluates it and creates an aggregated model from the submitted models. This aggregated model, along with the evaluation results, is then submitted to the upper-level organization. This process is repeated until the models and results reach H_i at the root of the tree. H_i collects and aggregates models from all sub-hospitals. If the performance of the aggregated model achieves the target performance, the training is terminated. Then, the models of the root nodes of other trees are aggregated to create the global model, and the process moves to the settlement stage. However, if the target performance is not achieved, model is redistributed to the tree, and training is repeated until the target performance is achieved. During this process, H_i reflects the training evaluation results in the virtual channel state with the patients. The updated state of the channel $S_1^{\gamma_{i,j,k,l}}$ can be represented as

$$S_1^{\gamma_{i,j,k,l}} : [H_{i,j} \rightarrow (y_{i,j,k,l} - \mathcal{I}_1^{i,j,k,l}), P_{i,j,k,l} \rightarrow (z_{i,j,k,l} + \mathcal{I}_1^{i,j,k,l})] \tag{12}$$

   where, $\mathcal{I}_1^{i,j,k,l}$ represents the incentive allocated to patient $P_{i,j,k,l}$ in round 1, and the updated state $S_1^{\gamma_{i,j,k,l}}$ indicates the transfer of funds amounting to $\mathcal{I}_1^{i,j,k,l}$ from $H_{i,j}$ to $P_{i,j,k,l}$.

3. *Settlement*: After all training process is completed, the settlement stage begins by submitting the off-chain channel states to the blockchain network. Channel settlement is conducted through virtual channel settlement by updating the state channel to its final state. This process proceeds sequentially from the channel factory of the upper nodes in the tree to the state channels of the leaf nodes in an up-bottom manner.

   First, after the completion of n rounds of training, when the state of the virtual channel with patient $S_n^{\gamma_{i,j,k,l}}$ is represented by

$$S_n^{\gamma_{i,j,k,l}} : [H_{i,j} \rightarrow (y_{i,j,k,l} - \mathcal{I}_n^{i,j,k,l}), P_{i,j,k,l} \rightarrow (z_{i,j,k,l} + \mathcal{I}_n^{i,j,k,l})] \tag{13}$$

   the updated state of the channels $S_1^{\beta_{i,j}}$ and $S_1^{\beta_{i,j,k}}$ between institution $H_{i,j}$ and the patients is represented as

$$S_1^{\beta_{i,j}} : [H_{i,j} \rightarrow (x''_{i,j} - \sum_{k=1}^{deg(i,j)} \sum_{l=1}^{deg(i,j,k)} \mathcal{I}_n^{i,j,k,l}),$$
$$H_{i,j,1} \rightarrow (x_{i,j,1} + \sum_{l=1}^{deg(i,j,1)} \mathcal{I}_n^{i,j,1,l}), ..., H_{i,j,k} \rightarrow (x_{i,j,k} + \sum_{l=1}^{deg(i,j,k))} \mathcal{I}_n^{i,j,k,l}]$$

$$S_1^{\beta_{i,j,k}} : [H_{i,j,k} \rightarrow (x'''_{i,j,k} - \sum_{l=1}^{deg(i,j,k)} \mathcal{I}_n^{i,j,k,l}),$$
$$P_{i,j,k,1} \rightarrow (x_{i,j,k,1} + \mathcal{I}_n^{i,j,k,1}), ..., P_{i,j,k,l} \rightarrow (x_{i,j,k,l} + \mathcal{I}_n^{i,j,k,l})]$$

4 Security Analysis

This section discusses the security properties that our proposed system features, including: *(1) Free-rider Attack:* In the process of creating a global model through the collaboration of multiple organizations, the proposed system ensures that all organizations collectively bear the cost of patient incentives across the entire system. This approach prevents the participation of organizations that do not actively engage in training or do not have a sufficient patient pool to contribute to the model, relying instead on a few large hospitals. This method reduces the burden of incentive payments for hospitals with a sufficient patient pool, motivating them to participate in the system. Additionally, it imposes a cost burden on free-rider attackers, thereby reducing the effectiveness of such attacks. *(2) Cost sharing:* In the proposed system, a channel factory is utilized to ensure that all organizations share the costs required for model development, regardless of the size of their sub-organizations. The channel factory allows the creation of sub-channels that divide the deposited funds among organizations according to an agreed-upon ratio. In our system, each organization deposits an equal proportion of funds into each sub-channel. The creation of sub-channels requires the digital signatures of all participants in the channel factory, preventing any single participant from arbitrarily creating sub-channels. Additionally, since payments cannot be refused at the termination of the channel, this approach ensures the fair distribution of incentives, thereby achieving our goal of equitable incentive allocation. *(3) Operational cost efficiency:* In the proposed system, a state channel architecture is utilized to minimize the computational costs associated with blockchain smart contracts. The channel incurs blockchain computational costs only during the creation and settlement phases. During the learning process, state updates and the creation of sub-channels occur off-chain, involving only the participants of the channel. Additionally, instead of using cryptocurrencies like Ethereum, the system employs tokens that represent real-world assets, thereby enhancing practicality(e.g. ERC-20, ERC-721, etc.).

5 Conclusion

We proposed a fair incentive distribution mechanism in a hierarchical system based on blockchain state channels. Our system is designed using state channels to ensure that multiple hospitals participating in federated learning contribute an equal budget to model development, regardless of their level of contribution to the training. The budgets provided by the hospitals are managed through state channels, ensuring that all hospitals share the incentive costs incurred during training equally. Additionally, the proposed system operates primarily off-chain, through communication between channel participants outside the blockchain, significantly reducing blockchain resource consumption and associated costs. This design also allows for flexible application of existing incentive mechanisms by enabling the evaluation of contributions through alternative incentive mechanisms.

6 Acknowledge

This research was supported as a 'Technology Commercialization Collaboration Platform Construction' project of the INNOPOLIS FOUNDATION (Project Number: 1711202494)

References

[1] S. Rani, A. Kataria, S. Kumar, and P. Tiwari, "Federated learning for secure IoMT-applications in smart healthcare systems: A comprehensive review," *Knowledge-Based Systems*, 2023.

[2] V. K. Prasad, P. Bhattacharya, D. Maru, S. Tanwar, A. Verma *et al.* "Federated learning for the internet-of-medical-things: A survey," *Mathematics*, vol.11, no.1, 2022.

[3] O. Rana, T. Spyridopoulos, N. Hudson, M. Baughman, K. Chard *et al.* "Hierarchical and decentralised federated learning," *in Proceeding of the 2022 Cloud Continuum*, pp.1-9, 2022.

[4] X. Wang, Y. Zhao, C. Qiu, Z. Liu, J. Nie *et al.* "Infedge: A blockchain-based incentive mechanism in hierarchical federated learning for end-edge-cloud communications," *IEEE Journal on Selected Areas in Communications*, vol.40, no.12, pp.3325-3342, 2022.

[5] G. He, C. Li, M. Song, Y. Shu, C. Lu *et al.* "A hierarchical federated learning incentive mechanism in UAV-assisted edge computing environment," *Ad Hoc Networks*, vol.149, 2023.

[6] Y. Zhao, Z. Liu, C. Qiu, X. Wang, F. R. Yu *et al.* "An incentive mechanism for big data trading in end-edge-cloud hierarchical federated learning," In Proceeding of the 2021 IEEE Global Communications Conference, pp. 1-6, 2021.

[7] H. Sun, X. Tang, C. Yang, Z. Yu, X. Wang *et al.* "HiFi-Gas: Hierarchical Federated Learning Incentive Mechanism Enhanced Gas Usage Estimation," In Proceedings of the AAAI Conference on Artificial Intelligence, vol.38, no.21, pp.22824-22832, 2024.

[8] S. Rahmadika, M. Firdaus, S. Jang, and K. H. Rhee, "Blockchain-enabled 5G edge networks and beyond: An intelligent cross-silo federated learning approach," Security and Communication Networks, 5550153, 2021.

[9] Y. Li, X. Wang, R. Zeng, M. Yang, K. Li *et al.* "VARF: An incentive mechanism of cross-silo federated learning in MEC," IEEE Internet of Things Journal, vol. 10, no. 17, pp. 15115-15132. 2023.

[10] M. Tang, and V. W. Wong, "An incentive mechanism for cross-silo federated learning: A public goods perspective," In Proceedings of the IEEE INFOCOM 2021-IEEE Conference on Computer Communications, pp. 1-10, 2021.

[11] M. Park, and S. Chai, "BTIMFL: A Blockchain-Based Trust Incentive Mechanism in Federated Learning," In Proceedings of the International Conference on Computational Science and Its Applications, pp. 175-185, 2023.

[12] C. Burchert, C. Decker, and R. Wattenhofer, "Scalable funding of bitcoin micropayment channel networks," Royal Society open science vol. 5, no. 8, 2018.

[13] S. Dziembowski, L. Eckey, S. Faust, J. Hesse, and K. Hostáková, "Multi-party virtual state channels," In Proceedings of the Advances in Cryptology–EUROCRYPT 2019: 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, pp. 625-656, 2019.

[14] J. Poon, and T. Dryja, "The Bitcoin Lightning Network: Scalable Off-Chain Instant Payments," 2016. [Online]. Available: https://lightning.network/lightning-network-paper.pdf