2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC), 24-28 September 2023, Bilbao, Spain, pp. 4840-4846
DOI: 10.1109/ITSC57777.2023.10422279

Federated Reinforcement Learning for Consumers Privacy Protection in Mobility-as-a-Service

Kai-Fung Chu and Weisi Guo

***Abstract*— Mobility-as-a-Service (MaaS) offers multi-modal transport modes in a single service platform, which requires tremendous data and software support. Among various types of data, consumers' data is vulnerable to the communication channel as it must be transmitted from the consumer end to the MaaS. Consumers put a high priority on the privacy of their data in selecting a service. This motivates the need for a secure information management system for MaaS to protect consumers' information from leakage. In this paper, we propose a federated reinforcement learning (FRL) approach for the information exchange intensive multi-modal journey planning process. The FRL approach protects the information from malicious information thieves by federating the global model training to a local one without sensitive information exchange while maintaining the same solution quality of enhancing MaaS profit and consumer satisfaction. We perform experiments on a test case based on New York City data. The results demonstrate that the FRL approach is effective in the MaaS multi-modal journey planning process. Compared to the baseline approaches, consumer satisfaction and MaaS profit increase by about 12% and 74%, respectively. This pilot study not only provides privacy protection insight into the MaaS multi-modal journey planning but also other privacy-concern applications.**

I. INTRODUCTION

Mobility-as-a-Service is a recent innovative transport concept that offers multi-modal transport modes in a single service platform [1]. Consumers may enjoy a seamless transport experience with one single entrance, including real-time transport information searching, journey planning, service pre-ordering, etc. MaaS has been operated in several countries such as the USA, UK, Canada, and Australia [2]. With MaaS, the transportation system in a city may have reduced traffic congestion, energy consumption, and air pollution [3]. Moreover, MaaS is a service provision model that can integrate not only existing transport modes but also future intelligent transportation systems applications such as traffic information forecasting [4], ride-sharing [5], and idle vehicles rebalancing [6]. Such a promising MaaS system requires a mature software system to handle the tremendous data among MaaS operators and consumers. The scheduler is one of the key components that process the journey planning query data repeatedly. Hence, the performance of the scheduler as well as the software system significantly affects consumer trust and acceptance of MaaS.

This work was supported by EPSRC MACRO - Mobility as a service: MAnaging Cybersecurity Risks across Consumers, Organisations and Sectors (EP/V039164/1)

Kai-Fung Chu and Weisi Guo are with the School of Aerospace, Transport and Manufacturing, Cranfield University, Bedford, MK43 0AL, UK kaifung.chu@cranfield.ac.uk; weisi.guo@cranfield.ac.uk

Security is one of the performance indexes of MaaS. For the centralized scheduler that processes sensitive data, such as consumer home and work address and commuting time, Insiders may attack the system to obtain unauthorized information [7]. The situation could be worse if the scheduler is intelligently designed using artificial intelligence (AI) to consider consumer behavior, experiences, and preferences for personal and unique transport services, which means more sensitive information such as occupation and age have to be processed [8]. Motivated by the privacy threat [9] and research gap in MaaS intelligent scheduler, we investigate countermeasures of this privacy threat to protect consumer privacy. Among the technologies, we believe that distributed processing could be a suitable technique to ensure the data is not disclosed to other participants. To do this, federated learning [10] that allows the training to be performed in distributed client agents by their private and local dataset could be integrated into the scheduling and reinforcement learning algorithms in MaaS. Each data owner may train their client model and securely share the learned gradient to update the server model without invasion of data privacy. Therefore, privacy is protected by restricting the interaction between the local environment and agents other than the environment owner. A federated MaaS platform could be a promising solution to address the aforementioned privacy issues.

In this paper, we investigate customer privacy protection against semi-honest participants in the journey planning process. To prevent information leakage, we transform the MaaS journey planning process into a federated architecture. The federation is built by dividing the centralized journey planning problem with consumer behavior and Markov decision process (MDP) formalization into sub-problems where each consumer is responsible for their own sub-problem. Based on the federation, a federated reinforcement learning (FRL) approach is presented to train a model that solves the problem without extensive information exchange. An equally weighted experience sampling mechanism is incorporated into the FRL to ensure the solution quality is similar to the centralized one. Experiments based on New York City dataset are conducted to evaluate the scenario under the FRL and other baseline approaches. The experimental results show that the FRL approach can maintain the same performance as the centralized one that increases consumer satisfaction and MaaS profit by about 12% and 74%.

The rest of this paper is organized as follows. Section II defines the MaaS journey planning problem and the threat models. Section III introduces the FRL approach for

TABLE I
NOTATION SUMMARY.

Notations	Meaning
$G(\mathcal{N}, \mathcal{A})$	Graph of the transport network
$\mathcal{F}$	Set of mobility providers
f	Mobility provider $f \in \mathcal{F}$
$\mathcal{N}$	Set of nodes in the network
$\mathcal{A}$	Set of links in the network
$\mathcal{A}_f$	Subset of $\mathcal{A}$ operated by $f \in \mathcal{F}$
$\mathcal{N}^+(i)$	Set of incoming locations of i
$\mathcal{N}^-(i)$	Set of outgoing locations of i
$\mathcal{K}$	Set of consumers
k	Consumer k
o^k	Origin node of consumer $k \in \mathcal{K}$
d^k	Destination node of consumer $k \in \mathcal{K}$
β_{ij}^f	Travel time cost of link $(i,j) \in \mathcal{A}_f$
δ_{ij}^f	Discomfort index of link $(i,j) \in \mathcal{A}_f$
ρ_{ij}^f	Price of link $(i,j) \in \mathcal{A}_f$
W^k	Utility weights of consumer k
w_β^k	Weight of β_{ij}^f
w_δ^k	Weight of δ_{ij}^f
w_ρ^k	Weight of ρ_{ij}^f
C_{ij}^f	Capacity of link $(i,j) \in \mathcal{A}_f$
x_{ij}^k	Binary variable for link $(i,j) \in \tilde{\mathcal{A}}_s$ that recommends to consumer $k \in \mathcal{K}$

MaaS. IV presents the experiment setting and results. Finally, Section V concludes this paper.

II. SYSTEM MODEL

In this section, we first define the threat model, multi-modal journey planning and consumer satisfaction problem. Table I summarizes the notation used.

A. Threat Model

We are interested in the scenario where all the participants, such as insiders of MaaS, mobility providers, and consumers, could be semi-honest, which is a common adversary model in privacy-preserving computation [11]. Some may also name it as honest-but-curious [12], which we may use these two terms interchangeably in this paper. This semi-honest model assumes that the participants are curious about any information they can obtain by following the protocol merely. In other words, they will not perform a malicious attack on the system, but they may try to infer sensitive information from the data they can access. Therefore, we can prevent unpremeditated information leakage under this threat model.

B. MaaS Problem Formulation

For the data processing part in MaaS, we explore the scenario where the largest amount of information is used by the scheduler, i.e., the MaaS multi-modal journey planning problem that considers consumer behavior, experiences, and preferences for personal and unique transport services. The two corresponding divided problems, the multi-objective journey planning problem and the consumer satisfaction problem, will be formalized in this section.

1) Multi-objective Journey Planning Problem: Consider a MaaS transport network modeled by a directed graph $G(\mathcal{N}, \mathcal{A})$ where $\mathcal{N}$ and $\mathcal{A}$ is the set of nodes and links in the network, respectively. We denote the set of mobility providers by $\mathcal{F}$ and each mobility provider $f \in \mathcal{F}$ provides transport services over their transport network $\mathcal{A}_f \in \mathcal{A}$. A transport service from i to j is provided on link (i,j) and the time cost, discomfort index, price, and operation cost of each service provided by f are represented by β_{ij}^f, δ_{ij}^f, ρ_{ij}^f, and μ_{ij}^f, respectively. The MaaS scheduler computes an optimal journey to be offered to the consumer based on the consumer's origin o^k and destination d^k and consumer behavior. Since it is a multi-modal journey planning problem for MaaS, the journey can be combined by multiple mobility providers. The optimization formulation can be referenced in the literature [13].

A binary decision variable, x_{ij}^{kf}, is defined in the formulation of the problem. x_{ij}^{kf} are used to represent the mobility service to be offered to the consumer:

$$x_{ij}^{kf} = \begin{cases} 1 & \text{if link } (i,j) \text{operated by } f \text{ offers to } k, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

The objective of the problem is to determine the optimal x_{ij}^{kf} such that the total utility cost of consumers are minimized:

$$\sum_{(i,j)\in\mathcal{A}_f, k\in\mathcal{K}, f\in\mathcal{F}} (w_\beta^k \beta_{ij}^f + w_\delta^k \delta_{ij}^f + w_\rho^k \rho_{ij}^f) x_{ij}^{kf}, \tag{2}$$

where β_{ij}^f, δ_{ij}^f, and ρ_{ij}^f are the travel time, discomfort index, and price of link $(i,j) \in \mathcal{A}_f$, respectively, and w_β^k, w_δ^k, w_ρ^k are the weighting of the corresponding utility terms.

Let $\mathcal{N}^-(i)$ and $\mathcal{N}^+(i)$ be the sets of outgoing and incoming locations of i, respectively, such that $\mathcal{N}^-(i) = \{j \in \mathcal{N} | (i,j) \in \mathcal{A}\}$ and $\mathcal{N}^+(i) = \{j \in \mathcal{N} | (j,i) \in \mathcal{A}\}$. To ensure the flow conservation in the transport network, the following equation requires to be imposed in the problem.

$$\sum_{j\in\mathcal{N}^-(i)} x_{ij}^{kf} - \sum_{j\in\mathcal{N}^+(i)} x_{ji}^{kf} = \begin{cases} 1 & \text{if } i = o^k, \\ -1 & \text{if } i = d^k, \\ 0 & \text{otherwise,} \end{cases} \quad \forall i \in \mathcal{N}, k \in \mathcal{K}, f \in \mathcal{F}, \tag{3}$$

Since the capacity of the transport service is limited, an equation is included to restrict the total number of consumers in the transport service:

$$\sum_{k\in\mathcal{K}} x_{ij}^{kf} \le C_{ij}^f, \quad \forall (i,j) \in \mathcal{A}_f, f \in \mathcal{F} \tag{4}$$

where C_{ij}^f is the capacity of transport service from i to j that operated by $f \in \mathcal{F}$.

As a whole, the multi-objective journey planning problem is given as follows:

Problem 1 (Multi-objective Journey Planning Problem):

$$\min_{x_{ij}^{kf}, y_{ij}^{f}} \quad (2)$$
$$\text{s.t.} \quad (3)\text{–}(4).$$

This problem is an integer linear program where a standard solver can solve if the weight of the objectives, $W^k = [w_\beta^k; w_\delta^k; w_\rho^k]$, are given. However, the utility weight W^k represents the preference for different objectives and it is hard to be defined. This is because they are difference for each consumer having different behaviors, experiences, and preferences. Therefore, another problem, namely, the consumer satisfaction problem, is formulated to determine the utility weight.

2) Consumer Satisfaction Problem: The objective of consumer satisfaction problem is to determine the set of utility weights that are best for consumer satisfaction and retention rate to the transport service based on the consumer profiles. As a result, the MaaS profit is expected to increase accordingly.

We model the consumer retention process as a 4-tuple MDP $\langle \mathcal{S}, \mathcal{A}, P, R \rangle$, where $\mathcal{S}$, $\mathcal{A}$, P, and R are the set of states and actions, state transition and reward function, respectively. The state $s_t^k \in \mathcal{S}$ is the concatenation of consumer satisfaction and consumer profiles. The action $a_t^k \in \mathcal{A}$ is the utility weight introduced in Problem 1, i.e., $a_t^k := \left[w_\beta^k; w_\delta^k; w_\rho^k\right]$ for time t and consumer k. $P(s_{t+1}^k | s_t^k, a_t^k)$ represents the satisfaction variation from states $s_t^k \in \mathcal{S}$ to $s_{t+1}^k \in \mathcal{S}$ for acting action $a_t^k \in \mathcal{A}$. $R(s_t^k, a_t^k, s_{t+1}^k)$ represents the total profit due to the transition from s_t^k to s_{t+1}^k after acting a_t^k.

The consumer satisfaction level in the state is represented by a N-level integer value, which also indicates the retention rate proportionally. Intuitively, the probability of consumers retains to the system is higher if they are satisfied with it.

We define H^k as the satisfaction level of consumer k. It is a variable affected on the transport service offered by the MaaS. In general, the satisfaction increases if the consumer is satisfied with the offered journey and decreases otherwise. The consumer satisfaction change is a function of the expectation difference on each utility:

$$H^k := \begin{cases} H^k + n & \text{if } E^k \geq \overline{E}^k, \\ H^k - n & \text{if } E^k \leq \underline{E}^k, \\ H^k & \text{otherwise,} \end{cases} \quad \forall k \in \mathcal{K}, \qquad (5)$$

where $\overline{E}^k$ and $\underline{E}^k$ is the upper and lower expectation difference threshold, n is the step size of the satisfaction level, and the expectation difference is defined as

$$E^k = \tilde{w}_\beta^k(\tilde{\beta}_{o^k d^k}^k - \beta_{o^k d^k}^k) + \tilde{w}_\delta^k(\tilde{\delta}_{o^k d^k}^k - \delta_{o^k d^k}^k) + \tilde{w}_\rho^k(\tilde{\rho}_{o^k d^k}^k - \rho_{o^k d^k}^k), \quad \forall k \in \mathcal{K}. \qquad (6)$$

where $\tilde{W}^k = [\tilde{w}_\beta^k; \tilde{w}_\delta^k; \tilde{w}_\rho^k]$ is the utility weight of consumer k, $\tilde{\beta}_{o^k d^k}^k, \tilde{\delta}_{o^k d^k}^k, \tilde{\rho}_{o^k d^k}^k$ are the utility expected implicitly by the consumer, and o^k and d^k are the origin and destination of consumer k, respectively. Expected utility is the utility of an ideal journey for the consumer. It can be determined by solving the planning problem as if the consumer is traveling alone in the system. The actual utility is the utility of the journey to be planned by MaaS, which can be deviated from the expected utility with an incompetent planner and limited capacity. Therefore, an incompetent planner may have a negatively large expectation difference on average.

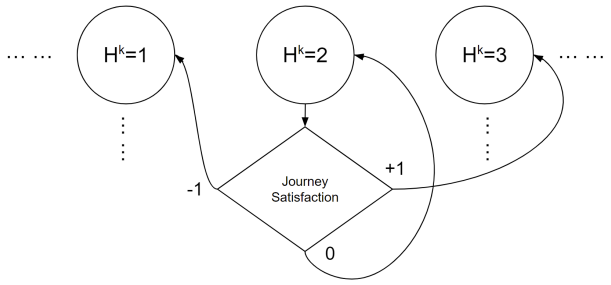

Fig. 1. Satisfaction transition representation.

A sample state transition representation of $n = 1$ is given in Fig. 1. The consumer experience of each journey may lead to satisfaction increases, decreases or unchanged, which affect the status and admission of the next journey. Similar consumer satisfaction models can be found in other field of studies such as supply chain [14] and product management [15].

Therefore, the consumer satisfaction problem can be formulated as:

Problem 2 (Consumer Satisfaction Problem):

$$\max_{a_t^k} \quad \sum_{k,t} R(s_t^k, a_t^k, s_{t+1}^k)$$
$$\text{s.t.} \quad (5)\text{–}(6).$$

With the MDP model, this problem can be solved by a reinforcement learning-based approach that trains by the transition and action of the consumers.

C. MDP model Components

1) State: The state s_t represents the information of the consumer, including the consumer's profile and satisfaction level. The profile is composed of the consumer's sensitive personal information, such as income and age. The satisfaction level can be considered as sensitive information to be protected. Therefore, the state of the consumer is the information we aim to protect.

2) Environment: We use the term environment to represent the state transition of the set of consumers in the MaaS. With different journey x_{ij}^{kf} planned by the MaaS scheduler, the environment may proceed to a different next state s_{t+1}^k and returns a reward r_t^k to the client agent at each time t. It can also be represented by transition probability $P(s_{t+1}|s_t, a_t)$.

3) Action: The action a_t^k represents the utility weight $\left[w_\beta^k; w_\delta^k; w_\rho^k\right]$ which indicates the weight of the terms of objective function: time, discomfort, and price in (2). The value of the weights affects the optimal journey to be planned in Problem 1.

4) Reward: The reward is a scalar feedback signal to indicate whether the agent is performing well. We use the reward r_t^k to represent the profit of MaaS for consumer k at time t, which is equivalent to the price minus the operating cost μ_{ij}^f, i.e,

$$r_t^k = \sum_{i,j,k,f} (\rho_{ij}^f - \mu_{ij}^f) x_{ij}^{kf}. \tag{7}$$

The objective of the MaaS scheduler is to maximize the reward. Intuitively, this could be achieved by offering the journey with the highest price and cost difference. However, inappropriate journeys may decrease the consumer satisfaction level and retention rate. Therefore, the scheduler should offer satisfied journeys to the consumers.

5) Agent: The agent represents the component for determining the action based on a given state. The agent aims to maximize the reward with an optimal action. Since the action is the utility weight, different actions affect the offered journey and thus the consumer satisfaction. Hence, the agent should learn to output utility weights that can increase the long-term reward. There are two types of networks in an agent: actor and critic. The actor network is used to output action based on the input state. A critic network is used to evaluate the performance of the state and action pairs.

6) Policy and value function: We use neural networks to approximate the policy and value function to be trained by the FRL algorithm. The neural network that approximates policy and value function is called the actor and critic network, respectively. To maintain the generality of the FRL approach, we use a standard deep fully-connected neural networks as the structure in this paper. Since the action is an array ranging from 0 to 1 so we use the $sigmoid$ as the activation function for the actor.

III. FEDERATED REINFORCEMENT LEARNING

In this section, we present the FRL algorithm that solves the problems introduced in Section II-B with privacy protection.

As discussed in Section II, the FRL algorithm is used to solve Problem 2. The MaaS aims to protect against information leakage during the actor and critic model training and inference. The MaaS scheduler is responsible for solving the Problem 2. In the centralized scenario, the MaaS scheduler would require the consumer to upload their profile and satisfaction level as the state $s_t^k \in \mathcal{S}$ for journey planning. This information could leak to semi-honest participants in MaaS if the algorithm is not privacy protected. Hence, we present the FRL algorithm to prevent information leakage under the semi-honest threat model assumption.

To avoid information leakage, the architecture is transformed to be a federated architecture, as shown in Fig. 2. In this architecture, the information and transition experience is stored locally in the consumer's device during the model training. The FRL algorithm is a federation based on the deep deterministic policy gradient [16] which is a model-free off-policy algorithm for determining the continuous action. So it matches our case where the utility weights are continuous values. In the federated architecture, each consumer builds a local client agent responsible for the individual utility weight. The neural network structure in the client and server are the same.

In each iteration, a set of consumers participate in the model inference and update. To ensure an unbiased sampling, the server sample the experience based on the buffer indices on behalf of clients instead of based on the participating consumer only. Each consumer holds a client experience replay buffer ER^k that stores the transition tuple $(s_t^k, a_t^k, r_t^k, s_{t+1}^k)$ for updating the client actors and critics. The reason for using an experience replay buffer to store transition for mini-batch sampling is to ensure the samples are independently and identically distributed. If the transitions are used to train the models immediately after being recorded, they would be sequential transitions, and become unstable.

The following update rules of the four models, namely, actor local, actor target, critic local, and critic target network, are applied to update the actor and critic based on a mini-batch sample with size B. The parameters of critic local θ^Q are updated based on the loss:

$$\frac{1}{B}\sum_i (r_i + \gamma Q'(s_{i+1}, \pi'(s_{i+1}|\theta^{\pi'})|\theta^{Q'}) - Q(s_i, a_i|\theta^Q))^2 \tag{8}$$

where i is the index of sample in the mini-batch and γ is the discount factor. The parameters of actor local θ^π are updated by computing the policy gradient:

$$\nabla_{\theta^\pi} J \approx \frac{1}{B}\sum_i \nabla_a Q(s, a|\theta^Q)|_{s=s_i, a=\pi(s_i)} \nabla_{\theta^\pi} \pi(s|\theta^\pi)|_{s_i}. \tag{9}$$

Critic target $\theta^{Q'}$ and actor target $\theta^{\pi'}$ are updated softly by assigning the parameters from the corresponding local networks with a factor of τ:

$$\theta^{Q'} := \tau\theta^Q + (1-\tau)\theta^{Q'}, \tag{10}$$

and

$$\theta^{\pi'} := \tau\theta^\pi + (1-\tau)\theta^{\pi'}. \tag{11}$$

Each participating client computes the gradients of actor and critic based on the update rules in each iteration. The mini-batch sample of each participating consumer is randomly selected by the server on behalf of the clients to enhance unbiased sampling. Each client selects samples according to the indices instructed by the server and computes gradients using their local experience only. As a result, all experience are stored in the client and not transmitted elsewhere. The gradients are then aggregated for updating the server actor and critic. Notice that in some federated learning algorithms, the server aggregates the model parameters instead of the gradient. However, aggregating the model parameters is not applicable in our case as we use Adam [17] as the optimizer, which contains a momentum term during the model update. Aggregating the model parameters updated by Adam using local experience makes the training unstable. The detailed algorithm is shown in Algorithm 1.

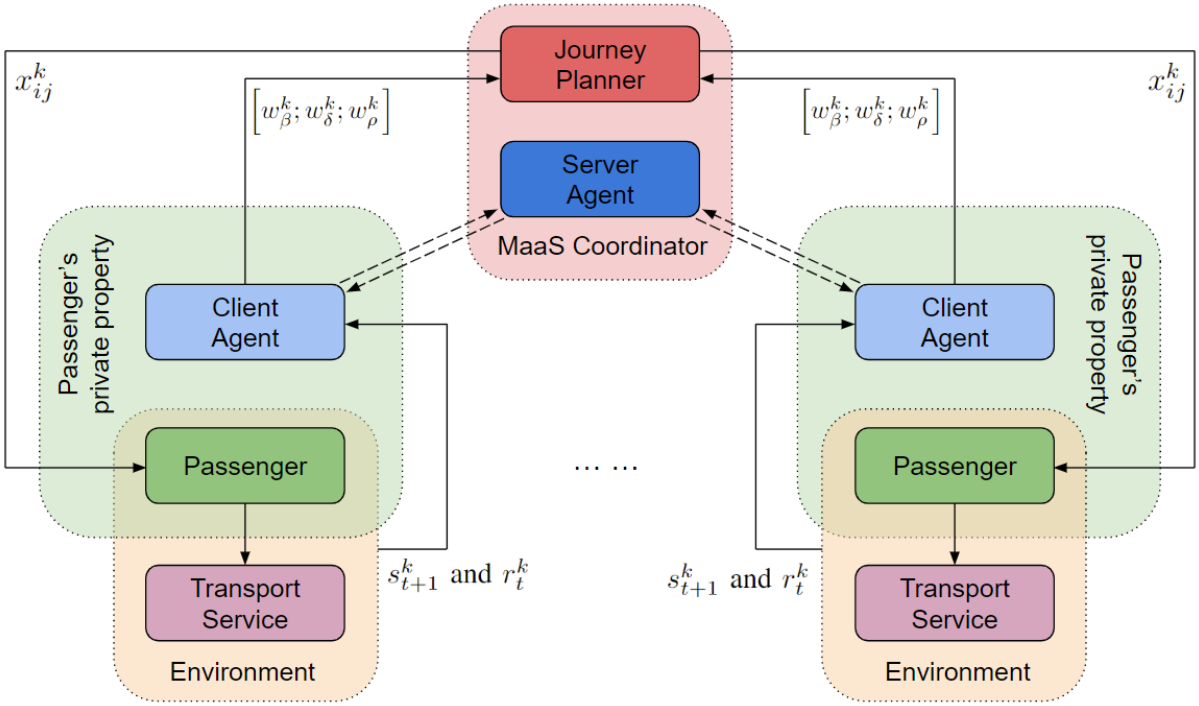

Fig. 2. Interaction between the environment, consumers and the MaaS coordinator.

IV. EXPERIMENTS

A. Experiment Setup

We present the experiments using the FRL approach to evaluate the performance. The experiments are based on real-world data from New York City (NYC). The graph of this NYC scenario is constructed based on the taxi zone maps[1] in the Manhattan region of NYC. The nodes in the transport network represent the taxi zone. We add an edge between two zones if the zones are neighbors on the map. Therefore, isolated zones without connected nodes are ignored and the corresponding data are filtered out. As a result, 63 zones in the map form a network in an irregular shape. 3 mobility providers are simulated to provide mobility services on each edges, resulting in 963 edges in the network. Each edge is associated with three utilities: time, discomfort, and price. The values are randomly generated between 0 and 1.

The set of consumers $\mathcal{K}$ is simulated based on another NYC datasets. The transport query and consumers' profiles are sampled from the NYC Taxi and Limousine Commission Trip Record Data[2] and Citywide Mobility Survey[3], respectively. The expected utility weight vector is calculated based on the profiles for Eq. (6) only, which is unknown to the MaaS scheduler throughout the experiments. Initial consumer satisfaction levels are 3. Problem 1 is solved by a standard optimizer in CVXPY [18] after the utility weights are obtained from the actor.

[1]https://data.cityofnewyork.us/Transportation/NYC-Taxi-Zones/d3c5-ddgc

[2]https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page

[3]https://www1.nyc.gov/html/dot/html/about/citywide-mobility-survey.shtml

TABLE II
PARAMETER SETTINGS.

Parameter	Definition	Value
$\lvert\mathcal{N}\rvert$	Number of nodes	63
$\lvert\mathcal{A}\rvert$	Number of links	963
$\lvert\mathcal{F}\rvert$	Number of mobility providers	3
$\lvert\mathcal{K}^m\rvert$	Number of consumers per episode	10
C_{ij}^f	Capacity	3
$\lvert\mathcal{R}\rvert$	Replay buffer size	10^6
B	Minibatch size	128
γ	Discount factor	0.99
τ	Target network soft update rate	0.001
-	Actor learning rate	0.0001
-	Critic learning rate	0.0003
-	Neural network optimizer	Adam
ϵ_0	Initial random explore rate	1
$\bar{\epsilon}$	Explore rate decay per episode	0.995
T	Number of iteration per episode	100
M	Number of episodes	2000
-	Number of neural network layers	3
-	Number of neurons of each layer	256
-	Range of satisfaction level	1 to 5
$\overline{E}^k$	upper expectation threshold	0.0
$\underline{E}^k$	lower expectation threshold	-0.1

Algorithm 1 Federated Reinforcement Learning

Initialize server actor local $\pi(s|\theta^{\pi})$ and critic local networks $Q(s,a|\theta^{Q})$ with parameters θ^{π} and θ^{Q}
Initialize parameters of server actor target $\pi'(s|\theta^{\pi'})$ and critic target networks $Q'(s,a|\theta^{Q'})$ with parameters $\theta^{\pi'} \leftarrow \theta^{\pi}$ and $\theta^{Q'} \leftarrow \theta^{Q}$
for $k = 1$ to $|\mathcal{K}|$ **do**
    Initialize consumer k parameters
end for
Broadcast θ^{π}, θ^{Q}, $\theta^{\pi'}$, and $\theta^{Q'}$ to client agents
for episode $= 1$ to M **do**
    for iteration $t = 1$ to T **do**
        Sample $\mathcal{K}_t^{in} \subseteq \mathcal{K}$ for model inference
        for consumer $k \in \mathcal{K}_t^{in}$ **do**
            $j^k \leftarrow$ random number between 0 and 1
            if $j^k < \epsilon$ **then**
                $a_t^k \leftarrow$ random vector between 0 to 1
            else
                $a_t^k \leftarrow \pi(s_t^k|\theta^{\pi})$
            end if
        end for
        Execute action $a_t = [a_t^1, \ldots, a_t^{|\mathcal{K}_t^{in}|}]$ to obtain new state $s_{t+1} = [s_{t+1}^1, \ldots, s_{t+1}^{|\mathcal{K}_t^{in}|}]$ and reward r_t
        for consumer $k \in \mathcal{K}_t^{in}$ **do**
            Store $(s_{i:i+N}^k, a_{i:i+N}^k, r_{i:i+N}^k, s_{i:i+N+1}^k)$
        end for
        Sample $\mathcal{K}_t^{up} \subseteq \mathcal{K}$ for model update
        if transitions in $\bigcup_{k \in \mathcal{K}_t^{up}} ER^k \geq B$ **then**
            Sample a mini-batch transitions with size B from $\bigcup_{k \in \mathcal{K}_t^{up}} ER^k$
            for consumer $k \in \mathcal{K}_t^{up}$ **do**
                Compute critic and actor gradient based on Eq. (8) and (9) for transitions in ER^k, respectively
            end for
        end if
        Update critic and actor local based on the aggregated critic and actor gradient, respectively
        Update critic and actor target based on Eqs. (10) and (11), respectively
        Broadcast θ^{π}, θ^{Q}, $\theta^{\pi'}$, and $\theta^{Q'}$ to clients
    end for
    $\epsilon := \epsilon\bar{\epsilon}$
end for

B. Experiment Results

We conduct experiments on three aspects: MaaS profit, Consumer satisfaction, and sampling mechanism. Two baseline approaches were evaluated with the FRL approach, namely, "random" and "fixed" approaches. The former sets the utility weights by random values between 0 and 1, while the latter always sets the utility weights to ones.

1) MaaS profit: To test the effectiveness of the FRL approach, we run the experiments with the two baselines for 2000 episodes, and each episode contains 100 iterations of transport queries of a different set of random consumers and evaluate the profit, which is the reward returned by the environment as discussed in Section II-C.4. The average profit against the episode is shown in Table III. Among the compared approaches, the FRL approach has the highest average profit 358.28 on average, and both "fixed" and "random" approaches are much worse than that. To see the trend during training, we plotted the moving average of profits in Fig. 3. From the figure, the profit of the FRL approach increases against the training episodes. This indicates that the FRL approach can learn from the experience and improve the policy gradually. Since the initial ϵ is high, most of the actions taken in the initial episodes are random. In the later episodes, ϵ decays to a small value, so the good FRL policy dominates the actions to be taken, further improving the policy for exploitation. For "random" and "fixed", they remain in a low reward level which indicates the incapability of adapting to consumers with different preferences.

TABLE III
AVERAGE REWARD OF DIFFERENT APPROACHES IN THE NYC SCENARIO.

Approach	Average reward (profit)
FRL	358.28
Fixed utility weights	236.77
Random utility weights	174.82

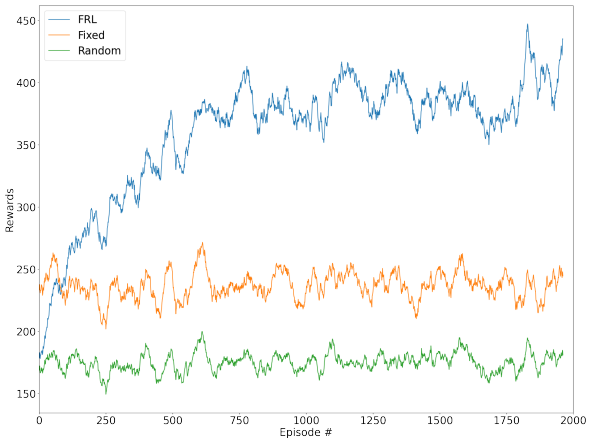

Fig. 3. Moving average of rewards of different approaches in the NYC scenario. The time window of the moving average is equal to 40.

2) Consumer Satisfaction: Consumer satisfaction shows us whether the consumer tends to retain using MaaS. Fig. 4 show the average satisfaction level of each episode. The color indicates the satisfaction level, as shown in the color bar on the right of the figures. The average satisfaction of the FRL, "fixed" and "random" approaches are 4.12, 4.06, and 3.28, respectively. The FRL shows a clear increasing trend at the beginning episode and then converges to around 4.12, which suggests the policies improve along with the episode. "fixed" and "random" remain in a lower satisfaction level. Therefore, a higher satisfaction level can increase the retention rate and also increase the profit.

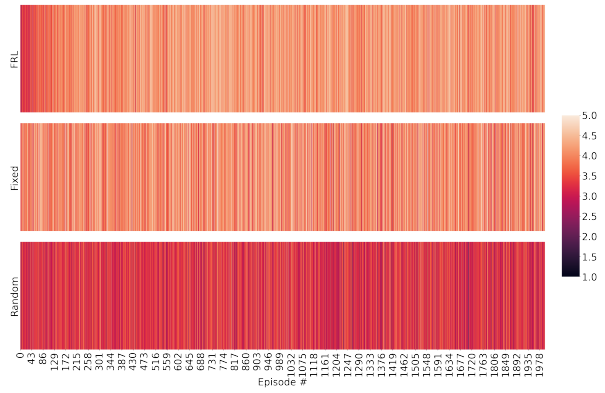

Fig. 4. Average satisfaction level of each episode in the NYC scenario.

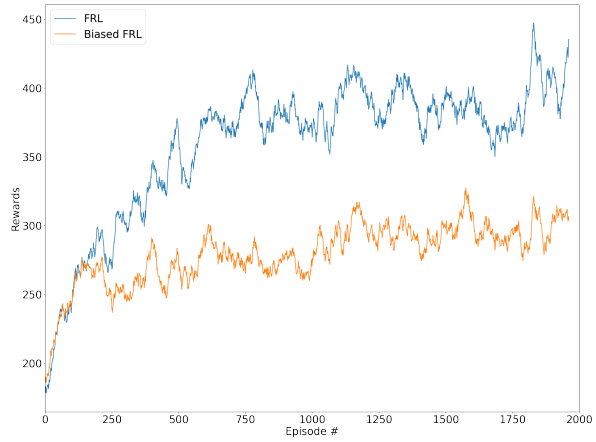

Fig. 5. Moving average of rewards of FDDPG approach and biased baseline in the NYC scenario. The time window of the moving average is equal to 40.

3) Sampling Mechanism: As discussed in Section III, an inappropriate sampling mechanism may significantly diminish the performance. To evaluate the effectiveness of the sampling mechanism in FRL, we compare the training with a biased baseline approach. The biased approach means that the experience is sampled from a consumer at a time. Fig. 5 shows the reward against the training episode of the FRL and the baselines. We can see from the figure that the FRL performs better than the biased baseline, which indicates the importance of an unbiased sampling mechanism. Without an unbiased sampling mechanism, bias could be introduced in federated training if a group of consumers frequently participate in the training than the rest of the consumers. Therefore, ensuring the sampling diversity as in the sampling mechanism of the FRL can enhance the performance.

V. CONCLUSIONS

The privacy threats endanger the popularity of MaaS. Curious participants may steal sensitive information through the privacy loophole. To protect the information, we proposed a federation architecture and an FRL approach that ensures the information is only accessible by its owner. Only intermediate results such as gradient are shared during the training. With this architecture, consumers can enjoy the MaaS service without being concerned about information leakage. The experiment results show that consumer satisfaction and MaaS profit increases by about 12% and 74%, respectively, using the FRL approach.

In the future, we may include additional cyber-security measures for MaaS to ensure information privacy in the client's local device. We may also extend the privacy-preserving architecture to other reinforcement learning-based approaches in transportation, such as the one in [19].

REFERENCES

[1] S. Hietanen, "Mobility as a service," *the new transport model*, vol. 12, no. 2, pp. 2–4, 2014.

[2] P. Jittrapirom, V. Caiati, A. M. Feneri, S. Ebrahimigharehbaghi, M. J. Alonso-González, and J. Narayan, "Mobility as a service: A critical review of definitions, assessments of schemes, and key challenges," *Urban Planning*, vol. 2, no. 2, pp. 13–25, 2017.

[3] A. Nikitas, K. Michalakopoulou, E. T. Njoya, and D. Karampatzakis, "Artificial intelligence, transport and the smart city: Definitions and dimensions of a new mobility era," *Sustainability*, vol. 12, no. 7, p. 2789, 2020.

[4] K.-F. Chu, A. Y. Lam, and V. O. Li, "Deep multi-scale convolutional lstm network for travel demand and origin-destination predictions," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 8, pp. 3219–3232, 2019.

[5] N. Agatz, A. Erera, M. Savelsbergh, and X. Wang, "Optimization for dynamic ride-sharing: A review," *European Journal of Operational Research*, vol. 223, no. 2, pp. 295–303, 2012.

[6] K.-F. Chu, A. Y. Lam, and V. O. Li, "Joint rebalancing and vehicle-to-grid coordination for autonomous vehicle public transportation system," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 7156–7169, 2022.

[7] F. Callegati, S. Giallorenzo, A. Melis, and M. Prandini, "Cloud-of-things meets mobility-as-a-service: An insider threat perspective," *Computers & Security*, vol. 74, pp. 277–295, 2018.

[8] K.-F. Chu and W. Guo, "Deep reinforcement learning of passenger behavior in multimodal journey planning with proportional fairness," *Neural Computing and Applications*, pp. 1–20, 2023.

[9] C. D. Cottrill, "Maas surveillance: Privacy considerations in mobility as a service," *Transportation Research Part A: Policy and Practice*, vol. 131, pp. 50–57, 2020.

[10] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Transactions on Intelligent Systems and Technology*, vol. 10, no. 2, pp. 1–19, 2019.

[11] J. Brickell and V. Shmatikov, "Privacy-preserving graph algorithms in the semi-honest model," in *Proc. Int. Conf. on the Theory and Appl. of Cryptology and Inform. Security*. Springer, 2005, pp. 236–252.

[12] Y. Aono, T. Hayashi, L. Wang, S. Moriai *et al.*, "Privacy-preserving deep learning via additively homomorphic encryption," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 5, pp. 1333–1345, 2017.

[13] G. Musolino, C. Rindone, and A. Vitetta, "Models for supporting mobility as a service (maas) design," *Smart Cities*, vol. 5, no. 1, pp. 206–222, 2022.

[14] C. Lam and W. Ip, "A customer satisfaction inventory model for supply chain integration," *Expert Systems with Applications*, vol. 38, no. 1, pp. 875–883, 2011.

[15] D. Martín-Consuegra, A. Molina, and Á. Esteban, "An integrated model of price, satisfaction and loyalty: an empirical analysis in the service sector," *Journal of Product & Brand Management*, 2007.

[16] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," in *Proc. Int. Conf. on Learn. Representations*, 2016.

[17] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[18] S. Diamond and S. Boyd, "CVXPY: A Python-embedded modeling language for convex optimization," *Journal of Machine Learning Research*, vol. 17, no. 83, pp. 1–5, 2016.

[19] K.-F. Chu, A. Y. Lam, and V. O. Li, "Traffic signal control using end-to-end off-policy deep reinforcement learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 7184–7195, 2022.

2024-02-13

Federated reinforcement learning for consumers privacy protection in Mobility-as-a-Serv

Chu, Kai-Fung

IEEE

Chu KF, Guo W. (2023) Federated reinforcement learning for consumers privacy protection in Mobility-as-a-Service. In: 2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC), 24-28 September 2023, Bilbao, Spain, pp. 4840-4846

https://doi.org/10.1109/ITSC57777.2023.10422279