sensors

MDPI

Article

Self-Interested Coalitional Crowdsensing for Multi-Agent Interactive Environment Monitoring

Xiuwen Liu, Xinghua Lei *, Xin Li * and Sirui Chen

College of Computer Science and Technology, China University of Petroleum (East China), Qingdao 266580, China; xwliu2015@whu.edu.cn (X.L.); z23070092@s.upc.edu.cn (S.C.)
* Correspondence: s21070073@s.upc.edu.cn (X.L.); lix@upc.edu.cn (X.L.)

Abstract: As a promising paradigm, mobile crowdsensing (MCS) takes advantage of sensing abilities and cooperates with multi-agent reinforcement learning technologies to provide services for users in large sensing areas, such as smart transportation, environment monitoring, etc. In most cases, strategy training for multi-agent reinforcement learning requires substantial interaction with the sensing environment, which results in unaffordable costs. Thus, environment reconstruction via extraction of the causal effect model from past data is an effective way to smoothly accomplish environment monitoring. However, the sensing environment is often so complex that the observable and unobservable data collected are sparse and heterogeneous, affecting the accuracy of the reconstruction. In this paper, we focus on developing a robust multi-agent environment monitoring framework, called self-interested coalitional crowdsensing for multi-agent interactive environment monitoring (SCC-MIE), including environment reconstruction and worker selection. In SCC-MIE, we start from a multi-agent generative adversarial imitation learning framework to introduce a new self-interested coalitional learning strategy, which forges cooperation between a reconstructor and a discriminator to learn the sensing environment together with the hidden confounder while providing interpretability on the results of environment monitoring. Based on this, we utilize the secretary problem to select suitable workers to collect data for accurate environment monitoring in a real-time manner. It is shown that SCC-MIE realizes a significant performance improvement in environment monitoring compared to the existing models.

Keywords: multi-agent reinforcement learning; self-interested coalition crowdsensing; environment monitoring; hidden confounder; worker selection

Citation: Liu, X.; Lei, X.; Li, X.; Chen, S. Self-Interested Coalitional Crowdsensing for Multi-Agent Interactive Environment Monitoring. *Sensors* **2024**, *24*, 509. https://doi.org/10.3390/s24020509

Academic Editors: Waheb Abdullah, AbdulRahman Alsewari and Mario De Oliveira

Received: 12 October 2023
Revised: 30 December 2023
Accepted: 3 January 2024
Published: 14 January 2024

1. Introduction

With the explosion of wireless communication and portable mobile devices, mobile crowdsensing (MCS) [1,2] has become a popular paradigm that appeals to workers to implement various sensing tasks and provide recommendation services [3,4]. However, different sensing environments and requirements make traditional MCSs very time consuming in accomplishing sensing tasks. Consequently, a highly effective tool, known as multi-agent reinforcement learning [5–11], has been developed to handle sequence recommendations in MCS. This tool demonstrates remarkable potential for addressing decision difficulties in unfamiliar sensing contexts. However, training in multi-agent reinforcement learning requires significant interactions and costs, which can lead to a decrease in data efficiency. Therefore, it is impractical to interact directly with the sensing environment. According to recent studies [12], environmental reconstruction is a viable technique that utilizes imitative learning to acquire environmental strategies from historical data. This approach not only increases the efficiency of interacting with virtual environments but also decreases interaction costs.

However, real-world scenarios exhibit a high level of complexity, so the data collected by workers from different sensory environments is significantly multi-source and

heterogeneous [8–11]. Figure 1 shows the variability of data collected in different sensing environments at different times. For example, data collected by workers from the target area tends to hold more significance compared to data collected from peripheral areas. Additionally, newly acquired data prove more valuable for environmental reconstruction than older data collected within the same sensing environment. Therefore, how to judge the importance of data and utilize it effectively is the first challenge of this paper.

Figure 1. Data collected at different times in different regions.

Note that it is difficult for workers to collect fully observable data in real time [13–16]. Most collected data have hidden confounding factors [17,18]. In addition, the next state relies on the former state and the action performed during training for multi-agent reinforcement learning. In real-world situations, however, the next state is more likely to be additionally influenced by hidden confounders. Similarly, hidden confounders affect the agents' actions and rewards as they interact with the sensing environment, which can reduce the accuracy of the environment monitoring. Hence, the second challenge is how to perform environmental monitoring in the presence of confounders.

Furthermore, rather than reactively waiting for given data, we are more interested in selecting the right workers to proactively sense important data for precise environmental monitoring. However, it is often complicated to forecast real-time data on workers in various sensing environments. Thus, the third challenge is how to select a group of workers to sense essential data in a real-time manner for accurate environmental monitoring.

To address the aforementioned difficulties, we have developed a unique framework called Self-interested Coalitional Crowdsensing of the Multi-agent Interactive Environment Monitoring (SCC-MIE). The objective of this framework is to acquire knowledge about the sensing environment and the underlying confounding factors utilizing partially seen data. It focuses on environment reconstruction and worker selection based on data confidence. In particular, we consider a reconstructor that comprises three representative agents: platform agent π_a, worker agent π_b, and confounding agent π_h, which leverages the available partial observations and historical data to reconstruct the missing or unobserved information. It aims to accurately model and understand the underlying structure of the sensing environment. Another discriminator is learned as a critic, outputting confidence values and judging whether the data samples are reliable or not. The reconstructor also uses the critic's judgment to improve its performance and challenges the critic by providing as little information as possible. Finally, we invert the selection of workers based on the confidence of the data in the environmental reconstruction, which in turn promotes continuous improvement in reconstruction accuracy. By forging cooperation between these two tasks, SCC-MIE could maximize the accuracy and reliability of the environment monitoring:

- We propose an efficient framework, called SCC-MIE, which consists of multi-agent imitation learning and a secretary-based online worker selection strategy. Based on data spatiotemporal heterogeneity and confounding effects, the former estimates the importance of the data and reconstructs the sensing environment. The latter aims to select workers to proactively sense critical data in an online manner to motivate environmental reconstruction.

- Considering confounding effects in real sensing environments, we design an imitation learning framework that includes confounder-embedded policy and a discriminator to learn the policy based on their interactions effectively.
- Extensive Evaluation: we conducted an extensive evaluation of the dataset using four different methods, which verified the validity of SCC-MIE.

2. Related Work

Mobile crowdsensing is a popular pattern that exploits groups of workers to perform a variety of sensing tasks in large-scale areas. To efficiently collect data, researchers have proposed numerous MCS schemes. For instance, in reference [19], the method employed dynamically priced tasks and social network effects to incentivize worker participation in MCS campaigns. Reference [20] used a set of UAVs to collect data from multiple mobile users to maximize geographic coverage and reduce the age of information (AoI) for all workers. Furthermore, crowdsensing platforms utilized polynomial-time greedy algorithms in task assignments to ensure fairness and energy efficiency among participating workers [4]. However, most existing works have overlooked the fact that the sensing data submitted by workers is often sparse and incomplete. As research progresses, compressed perception and matrix completion have emerged as practical options for data supplementation [10,11]. Compressed perception techniques recover the original data by observing low dimensional features, whereas matrix completion methods fill in missing values based on data structure and sample correlation. Applying these methods improves the handling of sparse and incomplete perceptual data.

Nevertheless, the present research has not thoroughly investigated the spatiotemporal correlation of perceptual data. In MCS environments, sensing data exhibit spatiotemporal correlations, wherein data from neighboring points in time and space may interact with each other. Therefore, considering this spatiotemporal correlation becomes crucial when performing state estimation to minimize resulting errors.

To address these issues, several recent studies have used reinforcement learning to establish a direct connection between spatiotemporal regions and their corresponding accuracy, to estimate their significance. Unfortunately, it is costly to run RL algorithms, which require much interaction with the environment to improve accuracy. As a result, environment reconstruction in RL has attracted widespread attention recently. Various studies have shown that imitation learning [21,22] enables environmental reconstruction by learning environmental strategies from past data. For instance, generative adversarial imitation learning (GAIL) [6] and its extension [23], multi-agent imitation learning (MAIL), simultaneously learn both strategies by defeating the discriminator that discovers the difference in the generated data from the real data. Furthermore, these workers are not directly applicable in practical situations. This is because the real world cannot provide a perfect sensing environment [14]. There is a high likelihood of confounders that affect the accuracy of the environment reconstruction [17].

Therefore, we present an efficient method that enables real-time environment reconstruction, keeping the importance estimates up to date, and selecting workers who can actively sense important data for environment reconstruction. The framework will combine reinforcement learning and imitation learning approaches to achieve accurate environment reconstruction in complex and incompletely observable real-world environments. In this way, we can overcome the challenges posed by the high cost and real-world complexity of traditional RL algorithms and provide an effective solution to environmental reconstruction in mobile crowdsensing.

3. Problem Definition and Framework Overview

3.1. Problem Definition

In this section, we define the MCS scenario and the sensing data. Then, our research question, worker selection, is determined, which leads to the focus of the article, namely environmental reconstruction. In a typical MCS campaign, each sensing task should

be executed in m target sensing regions with a duration of T. In this paper, we divide the duration T into t periods short enough to keep the sensed data stable and real time. Moreover, the main notations of this paper are listed in Table 1.

Table 1. List of important notations.

Notation	Explanation
i, N	Index of a worker, total of workers
t, T	Index of time slot, total of time slot
X_i^t	Sensory data of worker w_i at t-time slot
$\hat{X}$	Reconstructed data for X
O, U	Observable data set and unobservable data set
$\tau = \{\tau_1, \tau_2, \ldots, \tau_n\}$	Sensing the trajectory of workers.
c_i	Cost of collecting data.
B	Budget of collecting data.
P, p_i	Confidence level, the judgment of the discriminator D

Next, we define the trajectory and sensing data. The research question of this paper is defined according to them.

Definition 1. *(Trajectory) A trajectory is a chronological sequence of spatiotemporal points. In addition, n workers collect trajectories in time T, denoted as* $\tau = \{\tau_1, \tau_2, \ldots, \tau_n\}$*, where each point* $\tau_i = (x, y, t)$ *is made up of positions* (x, y) *(i.e., longitude and latitude) at the t-time slot.*

Definition 2. *(Sensing Data) For each sensing task, we consider a set of workers, denoted as* $w_i = \{w_1, w_2, \ldots, w_n\}$*, each worker* $i \in N$ *with a cost* c_i *independently determines its sensory data* X_i^t *at t-time slot.*

We assume a set of tasks with m sensory regions, n time periods, a budget of B, and a duration of T. In this paper, our primary objective revolves around the precise reconstruction of the environment while minimizing the discrepancy between the generated data and real-world data. To achieve this goal, we focus on selecting workers who will contribute their individual sensing data to the MCS campaign. The decision-making process for the platform involves assessing the estimated importance of each worker's data and considering the remaining budget, denoted as B, to determine whether to include a worker's data for the task of environment reconstruction.

$$minimize \sum_{t=1}^{T} D\left(X_t, \hat{X}_t\right) = \left(X - \hat{X}_t\right)^2 \tag{1}$$

$$subject\ to\ \hat{X}_t = G(X_t),\ w \subseteq W,\ \sum_{w_i \in w} c_i \leq B \tag{2}$$

In our analysis, we assume that the addition of new data leads to a reduction in error and transforms the objective function into a maximization problem. However, the challenge lies in the worker selection problem, which essentially involves choosing a subset of k elements to maximize an ensemble function [24]. Taking into account the intricate nature of environment reconstruction and the dynamically evolving real-time scenarios, we recognize that the main problem addressed in this paper falls under the realm of NP-hard problems.

3.2. Framework Overview

In this paper, we propose the SCC-MIE, which consists of two key components: worker selection and environmental reconstruction. Figure 2 illustrates the overall architecture of SCC-MIE, showing the interaction between these two key elements.

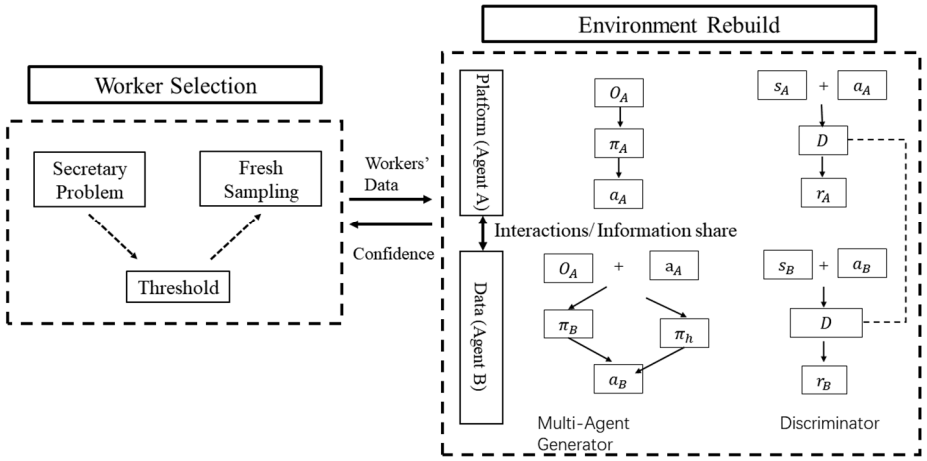

Figure 2. System framework SCC-MIE.

Worker Selection: this component determines whether or not to select the worker based on the quality of the data. Firstly, sampling from historical trajectories. Then, importance estimation is derived from the spatiotemporal data of the sample (workers). Ultimately, a secretary strategy is adopted to select suitable workers, aiming to sense important data for environment reconstruction proactively.

Environment Reconstruction: we develop an online framework for reconstructing the environment using spatiotemporal input from workers. This framework consists of a reconstructor and a discriminator. Furthermore, SCC-MIE forges cooperation through information sharing among agents to boost their performance. First, the reconstructor extracts the spatiotemporal embedding of the data and then reconstructs the environment by considering the ignored confounders. The discriminator receives additional information from the reconstructor and computes confidence measures for the estimation results by observing the reconstructed data.

4. Online Multi-Agent Environment Reconstruction

Due to the hidden confounding factors in realistic sensing environments, we introduce a novel online multi-intelligent environment reconstruction method, namely SCC-MIE (self-interested coalitional crowdsensing for multi-agent interactive environment monitoring). Our approach aims to tackle two interconnected tasks simultaneously. The first task involves utilizing a reconstructor G with generative capabilities to perform environmental reconstruction by generating an approximation $\hat{X}$ based on partial observations X. The second task leverages a discriminator D to calculate an interpretable confidence level P, incorporating information from X and $\hat{X}$. This parallel undertaking enables us to solve the complexities of environment monitoring effectively in the presence of hidden confounders.

The two tasks can be formalized as follows:

$$A : G(X) = \left(\hat{X}\right), \mathcal{L}_{\mathcal{A}} = loss_A\left(X, \hat{X}\right) \tag{3}$$

$$B : D\left(X, \hat{X}\right) = \left(X - \hat{X}\right)^2 = P \tag{4}$$

Considering the cooperative and interactive nature of the two tasks defined in Equations (3) and (4), an intuitive solution is adversarial learning, such as a generative adversarial network (GAN), which not only enables mutual learning between the generator and the discriminator but also improves the performance of both tasks. As a result, we want to minimize the following loss function:

$$\underset{G}{\text{argmin}}\text{argmax}_{D\in 0,1} E_{\mathbf{X}\sim p_E}[\log D(x)] + E_{z\sim p_Z}[\log(1 - D(G(z)))] \tag{5}$$

where p_z is a distribution, the generator is responsible for generating samples (in our case, $\hat{X}$) that approximate the desired data distribution p_E. However, it is essential to note that the training of the primary task heavily relies on the output of the discriminator D, which can result in the potential loss of essential information. Furthermore, solving the minimax optimization problem involved in GAN-style models is inherently more challenging compared to directly minimizing the loss function.

In addition, we explore an alternative approach called generative adversarial imitation learning (GAIL), which has gained popularity as a method for imitation learning. Unlike traditional imitation learning, which directly learns policies from expert demonstrations and has shown practical benefits, GAIL takes a different approach. It leverages the concept of generative adversarial networks to learn policies by competing against a discriminator network trained on both expert demonstrations and generated trajectories. This approach offers a distinct perspective on imitation learning and has shown promise in various applications.

The problem at hand can be formulated as follows: we aim to train a policy π that minimizes the loss function $l(\mathrm{s},\pi(\mathrm{s}))$ under the discounted state distribution of the expert policy $P_{\pi_e}(s) = (1-\gamma)\sum_{t=0}^{T}\gamma^t p(s_t)$. The objective of imitation learning is denoted as $\pi = \mathrm{argmin} E_{s\sim P_{\pi_e}}[l(s,\pi(s))]$ However, traditional imitation learning approaches have certain limitations, such as the instability of behavioral cloning and the difficulty of operationalizing reverse reinforcement learning. To address these drawbacks, studies have shown that generative adversarial imitation learning (GAIL) can achieve comparable theoretical and empirical results while being more efficient and avoiding the pitfalls of traditional imitation learning. GAIL employs a GAN framework, where a policy generator G is guided by a reward function represented by a discriminator D. The objective function of GAIL is defined as follows:

$$\underset{\pi}{\mathrm{argmin}}\,\mathrm{argmax}_{D\in 0,1} E_{\pi}[\log D(s,a)] + E_{\pi_E}[\log(1-D(s,a))] - \lambda H(\pi) \tag{6}$$

Here, $H(\pi) \triangleq E_{\pi}[-log\pi(a \mid s)]$ denotes the entropy of the policy π, and p_E represents the joint distribution of experts over state-action pairs. This formulation allows GAIL to derive a policy from expert examples efficiently.

Indeed, GAIL works to improve similarity from generated trajectories to expert trajectories. In this way, the learned strategy is executed in the environment and the update is performed using the gradient descent method. The loss function used for policy updating is as follows:

$$\begin{aligned} &l(s,\pi(s)) = E[\log D(s,a)] - \lambda H(\pi) \cong \\ &E_{\tau_i}[\log \pi_{a|s} Q(s,a)] - \lambda H(\pi) \end{aligned} \tag{7}$$

where $Q(s,a) =_{\tau_i} [log(D(s,a))|s_0 = s, a_0 = a]$ represents the state-action value function.

It quantifies the expected log-probability assigned by the discriminator to a given state action pair based on the trajectories τ_i. In recent developments, GAIL has demonstrated its effectiveness in environmental reconstruction. This extension leverages the collaborative efforts of multiple agents, allowing for enhanced performance and improved reconstruction outcomes.

In this study, we deploy the SCC-MIE framework to construct a virtual environment incorporating hidden confounders. This is achieved by historical data comprising observable information as well as unobservable confounding factors. By incorporating both types of data, we create a comprehensive and realistic virtual environment to capture the complexity and interactions of the underlying system. We then consider an interactive system with three agents on the basis of GAIL, representing worker policy π_B, platform policy π_A, and confounder policy π_h. We find that worker and platform policies are "mutual environments" from the MDP perspective. The platform's observations are the workers' reactions and the platform's actions are recommendations to the workers. Correspondingly, the observation data of workers is the platform's recommendation, and their actions are the workers' responses to the platform. In the environment, hidden confounders generate

dynamic effects that affect actions made by the platform and the worker. In other words, these agents interact with each other while all are affected by hidden confounding factors. Specifically, the platform utilizes the workers' spatiotemporal data to gain the recommendation of workers' next action via strategy π_A. Then, the worker derives a response for the next temporal state based on the observed data, recommendation, and unobserved confounders via strategy π_B.

Furthermore, we incorporate the dynamic impact of the hidden confounder H by modeling it as a hidden policy denoted as π_h. The main objective of this paper is to leverage the observed data, specifically the trajectories $\{x, a_A, a_B\}$, to imitate the strategies employed by agents A and B. The influence of the confounders is captured by inferring the hidden strategies. The objective function for multi-agent imitation learning is formulated as follows:

$$\arg \min_{(\pi_a, \pi_b, \pi_b)} E_{s \sim P_{\tau_{real}}}[L(s, a_A, a_B)] \tag{8}$$

where a_A and a_B are dependent on three policies. According to Equation (7), we apply the GAIL framework to obtain the following:

$$\begin{array}{l} L(s, a_A, a_B) = E_{\pi_a, \pi_h, \pi_b}[\log D_a(s, a_A) D_{hb}(s, a_A, a_B) - \\ \lambda \sum_{\pi \in \{\pi_a, \pi_h, \pi_b\}} H(\pi) \\ = E_{\pi_a}[\log D_a(s, a_A)] - \lambda H(\pi_a) + \\ E_{\pi_h, \pi_b}[\log D_{hb}(s, a_A, a_B)] - \lambda \sum_{\pi \in \{\pi_h, \pi_b\}} H(\pi) \\ = l(s, \pi_a(s)) + l((s, a_A), \pi_b \circ \pi_h((s, a_A))) \end{array} \tag{9}$$

which demonstrates that the optimization process can be discretized into two components: an optimization strategy π_a and a joint strategy $\pi_{hb} = \pi_b \circ \pi_h$.

During the collaborative process, when the reconstruction error stops providing valuable information for the judgment of discriminator D, we then conclude that the reconstructor achieves a satisfactory data reconstruction performance. Thus, Equation (9) essentially minimizes $l(s, \pi_a(s))$ and $l((s, a_A), \pi_b \circ \pi_h((s, a_A)))$, respectively, to output different confidence levels for the data to optimize the reconstruction performance continuously. However, it can be observed that the process of minimizing the loss function increases the reconstruction error for observable samples and decreases the reconstruction error for unobserved confounders. We, therefore, reweight the loss function, i.e., according to the observable data (O) and the unobservable confounding variables (U), using the confidence level.

$$Re - weighting\ factor = \begin{cases} 1 + 1/p_i, \ x_{i \in O} \\ -1/(1 - p_i), \ x_{j \in U} \end{cases} \tag{10}$$

In Equation (10), p_i represents the discriminator's judgment on whether the observation x_i is considered observed or not. This equation highlights the distinct behavior of reconstruction errors for observed data and unobserved confounders. Notably, if the discriminator D determines that an observation x_i from the set of observed data O is unobservable or unreliable (indicated by a confidence value $p_i \to 0$), the corresponding reconstruction error will be noticeably larger. This emphasizes the importance of reliable observations in achieving accurate reconstruction results.

Based on the reweighting function, we materialize the loss function, which is expressed as follows:

$$Loss(x) = \begin{cases} l(s, \pi_a(s)), x_{i \in o} \\ l((s, a_A), \pi_{hb}((s, a_A))), x_{j \in U} \end{cases} \tag{11}$$

where

$$l(s, \pi_a(s)) \cong \mathbb{E}_{\tau_i}[\log \pi_a(a_A \mid s) Q(s, a_A)] - \lambda H(\pi_a) \tag{12}$$

and

$$l((s,a_A),\pi_{hb}((s,a_A))) \cong E_{\tau_i}[\log \pi_{hb}(a_B|s,a_A)Q(s,a_A,a_B)] \\ -\lambda\sum_{\pi\in\{\pi_b,\pi_b\}} H(\pi) \tag{13}$$

Moreover, $Q(s,a_A) = \mathbb{E}_{\tau_i}[log(D((s,a_A))|s_0 = s, a_{A_0} = a_A]$ and $Q(s,a_A,a_B) = \mathbb{E}_{\tau_i}[log(D((s,a_A),a_B))|s_0 = s, a_{A_0} = a_A, a_{B_0}]$ denote the state-action value functions for policy π_a and policy π_{hb}, respectively.

By extending the GAIL framework, we accomplish the objective of imitating the strategies employed by each agent. This extension leads to the development of SCC-MIE to address the challenges of hidden confounders in reconstructing the environment using multi-agent methods.

5. Detailed Model Construction

In the context of environment reconstruction, we address the problem by introducing a reconstructor and a discriminator, which capture the dependencies within the data to facilitate the reconstruction process. Figure 3 illustrates the workflow.

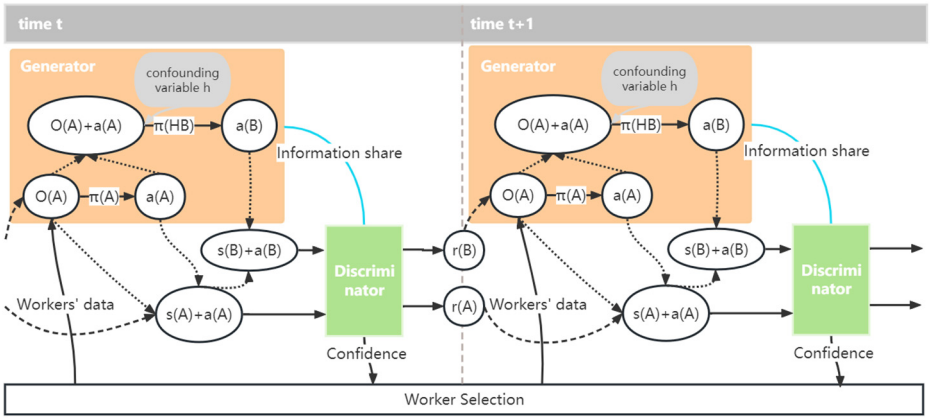

Figure 3. Workflow diagram of SCC-MIE.

5.1. Confounder Embedded Policy

In our framework, the reconstructor is responsible for inferring the hidden confounders and recovering the unobservable data, while the discriminator evaluates the quality and authenticity of the reconstructed environment. Thus, we assume that the data of agent A and agent B are observable and transparent. However, the data of agent H is unobservable, due to the presence of hidden confounders. As a consequence, we design a joint policy $\pi_{hb} = \pi_h \circ \pi_b$, which incorporates the confounder policy π_h and policy π_b. In other words, the joint strategy can be expressed as $\pi_{hb}(o_A, a_A) = \pi_b(o_A, a_A, \pi_h(o_A, a_A))$, where inputs o_A, a_A, and output a_B can be obtained from historical data. To summarize, we formulate the environmental reconstruction problem as a Markov decision process, using an imitation learning approach to train both policies by imitating observed interactions.

For the policies update in the generator, the joint policy π_{hb} utilizes reward r_{HB}, which receives from discriminator D, to update. The policy π_A updates by reward r_A. In SCC-MIE, generating the hidden strategy π_h is a byproduct arising as the policy π_{hb} and the policy π_A are continuously optimized. Such a hidden policy better represents the impact of confounders present in real environments on workers and the platform. Thus, through these two phases, the concealed policy π_h is iteratively and indirectly optimized to restore the actual confusing impact as much as possible. In order to make the training process faster and more stable, we adopt PPO, which is more efficient in terms of samples than other policy optimization algorithms, to update the above policy.

PPO is a widely used reinforcement learning algorithm that belongs to the policy optimization-based approach. Unlike other algorithms based on policy optimization, its core idea is to optimize the policy while limiting the difference between the new policy and the old one to ensure training stability. To encourage strategy improvement while

penalizing excessive strategy updates, a specific objective function is introduced. PPO is adept at handling continuous action spaces and performs well in many tasks without requiring complex tuning parameters. These advantages make PPO an efficient, stable, and general algorithmic alternative for reinforcement learning.

In this framework, we develop a discriminator compatible with both classification tasks for π_{hb} and π_a. The discriminator takes as inputs the state-action pairs (o_A, a_A, a_B) and $(o_A, a_A, 0)$. Firstly, the discriminator classifies the true and generated state-action pairs from π_{hb}. It then proceeds to classify the state-action pairs from π_a. In contrast to typical generative adversarial learning frameworks where only one task is modeled and learned in the generator, this paper defines the loss functions for π_{hb} and π_a as follows:

$$Loss = \begin{cases} loss(\pi_{hb}) = E_\tau[log(D_\sigma(o_A, a_A, a_B))] + E_{\tau_{real}}[log(1 - D_\sigma(o_A, a_A, a_B))] \\ loss(\pi_a) = E_\tau[log(D_\sigma(o_A, a_A, 0))] + E_{\tau_{real}}[log(1 - D_\sigma(o_A, a_A, 0))] \end{cases} \tag{14}$$

The probability that the input pairs come from real data is the output to the discriminator. This paper labels the true state-action pairs as 1 and the generated false state-action pairs as 0. The discriminator is trained through supervised learning using these labels. The discriminator is trained for supervised learning by this conception. It is then used as the policy reward while simulating the interaction. The reward function of the policy π_{hb} and the policy π_a can be expressed as follows:

$$Reward = \begin{cases} r^{HB} = -\log(1 - D(o_A, a_A, a_B)) \\ r^{A} = -\log(1 - D(o_A, a_A, 0)) \end{cases} \tag{15}$$

The confounder-embedded policy plays a crucial role in capturing the influence of hidden confounders, enabling the framework to model and reconstruct the environment accurately. By integrating this policy into the learning process, SCC-MIE aims to achieve robust and accurate reconstruction results, even in unobservable factors.

During the process of environment reconstruction, we begin by formulating the environmental reconfiguration problem as a Markov decision process (MDP) quintuple denoted by (S, A, P, R, γ). In this quintuple, the symbol S represents the state space and represents the action space. The function $P : S \times A \mapsto S$ signifies the state transition probability model, and $R : S \times A \mapsto R$ denotes the reward function. Additionally, γ serves as the discount factor for cumulative reward. Subsequently, a trajectory is randomly sampled from the observation data, with the initial observation o_A assigned as the observation state for the first time slot. To generate a complete trajectory, the policies π_a and π_b are employed, triggered by the initial observation o_0^A. Utilizing the policy π_a, the action a_t^A is determined based on the observation o_t^A. Similarly, the joint strategy π_{hb} guides the selection of the action a_t^B. Equation (15) represents simulated rewards used for updating the strategy during the adversarial training phase. Moving forward, given the observation o_t^A and the action a_t^B, the subsequent observation o_t^{A+1} can be obtained using predefined transfer probabilities. This step is repeated until reaching the end state, resulting in the generation of the pseudo-trajectory.

Algorithm 1 elucidates the process of environment reconstruction, utilizing the generative adversarial training framework. The generator takes center stage within each iteration of this algorithm, orchestrating simulated interactions by employing the policy π_a and the policy π_{hb}. These interactions lead to the assembly of a trajectory set denoted as π_{sim}, encompassing the line 5 to 17 of the algorithm. Subsequently, the policy π_a and the policy π_{hb} undergo iterative updates using the proximal policy optimization (PPO) technique, utilizing the generated trajectories π_{sim}, as a starting point. This transformative process unfolds within line 18 of the algorithm. Guided by the passage of K generator steps, the compatible discriminator assumes its rightful place. In line with the orchestration of Algorithm 1, the compatible discriminator undergoes a two-step training regimen, unfolding within line 20. Notably, the predefined transition dynamics, nestled within line 13, are intricately tied to the specific tasks at hand. These dynamics mold each step of

the reconstruction process. SCC-MIE adeptly emulates the observed interaction policies, transcending the confines of mere observation to recover the concealed confounder that lies beyond.

Algorithm 1: SCC-MIE algorithm

Input: Trajectory data $D_{real} = \{\tau_1, \tau_2, \cdots, \tau_n\}$.
Output: π_a, π_b, π_h.
Initialization policies π_{hb} and π_a with parameters θ_{hb} and θ_a, and discriminator D with parameter σ;
for $i = 1, 2, \ldots$ **do**
  for $k = 1, 2, \ldots, K$ **do**
    $\tau_{sim} = \varnothing$;
    for $j = 1, 2, \ldots, N$ **do**
      $\tau_j = \varnothing$;
      Select a random trajectory τ_r from D_{real}
      Set the first state to the initial observation o_0^A;
      for $t = 0, 2, \ldots, T-1$ **do**
        Simulate the actions a_t^A, a_t^B by the policy π_a and the policy π_{hb}, respectively
        Calculate the rewards r_t^A and r_t^{HB} by Equation (15);
        Derive the next observation o_{t+1}^A
        Insert $\{o_t^A, a_t^A, a_t^B, r_t^A, r_t^{HB}\}$ into the trajectory τ_j
      end for
      Integration of the computed data
      end for
      Update parameters θ_{hb} and θ_a according to PPO;
  end for
  Update the discriminator D by minimizing the losses;
end for

5.2. Worker Selection

In the dynamic real world, worker participation is real time and random. Faced with unknown information, this paper decides whether to select it or not based on the importance (confidence) of the data, i.e., workers who contribute high-quality data are selected and workers with low-quality data are discarded. In addition, we simplify the worker selection problem by viewing it as choosing the best one out of w workers. In addition, we simplify the worker selection problem by viewing it as choosing the best one out of w workers. In this case, the worker selection problem is a typical secretarial problem. To better solve this problem, we first construct a sample set of workers and observe and eliminate the top $1/e$ workers to understand their utility distribution. Then, in practice, we use historical data to learn about this distribution in real time and then select the best worker from the known distribution.

To streamline the process and harness the potential of forthcoming workers, we introduce a novel sampling methodology, aptly named Algorithm 2. This ingenious approach enables us to make informed decisions by capitalizing on historical data. The algorithm begins by extracting a sample set of size $j-1$ from the wealth of past records (line 1). Subsequently, we meticulously evaluate the utility of each upcoming worker against the pinnacle of utility, also known as the threshold, within the sample set. This critical assessment dictates whether a worker is selected for further consideration (line 3). When the current worker is not chosen, they are gracefully assimilated into the sample set, seamlessly replacing one of its members through a random resampling process (line 5–line 6). Importantly, it should be noted that the threshold evolves dynamically over time as we progressively eliminate workers with the highest utility. Consequently, our endeavor entails creating a sample set comprising $j-1$ workers, emulating a near approximation of the discarded workforce. This astute strategy allows us to gain valuable insights into estimating the threshold for every new worker, laying the foundation for informed decision making.

Algorithm 2: Worker selection algorithm

1: **Input**: selected workers μ, new workers ω_i, time: T; budget: B
2: Conduct X from history data according to t and T.
3: **while** $c_i <= B$ **do**
4: if $D(\mu, \omega_i) >= \max\{X\}$ **then**
return $\mu \cup w_i$;
5: **else**
6: $X \cup D(\mu, \omega_i)$;
7: Select z at random from X, X=X | {z};
8: ω_i++
9: **end if**
10: **end while**

6. Performance Evaluation

The AIMSUN dataset is a virtual collection of vehicle trajectories that has been constructed using a microscopic traffic simulation model. The system utilizes a dynamic traffic assignment method to suggest appropriate routes for individual vehicles [25]. Next, we analyze five distinct models: binomial, C-logit, proportional, polynomial logit, and fixed. The several route models in the AIMSUN dataset offer alternative methods for determining probability and making judgements when selecting routes for vehicles. The fixed model employs a greedy approach that prioritizes minimizing the trip time for each origin–destination (OD) pair. By examining these route models, one can assess the performance of SCC-MIE in comparison to various tactics. This evaluation offers valuable insights into the usefulness of SCC-MIE and allows for a comparison with the baseline models in addressing route selection difficulties in the AIMSUN dataset.

Table 2 presents a partial set of data, and Figure 4 is a network diagram of three routes based on the data in Table 2.

Table 2. Simulated traffic network partial data in AIMSUN environment.

OriginID	Number	SectionID
1	1	252
1	2	247
1	3	275
1	4	329
1	5	398
1	6	442
2	1	252
2	2	249
2	3	326
2	4	329
2	5	398
2	6	442
3	1	252
3	2	249
3	3	326
3	4	327
3	5	371
3	6	442

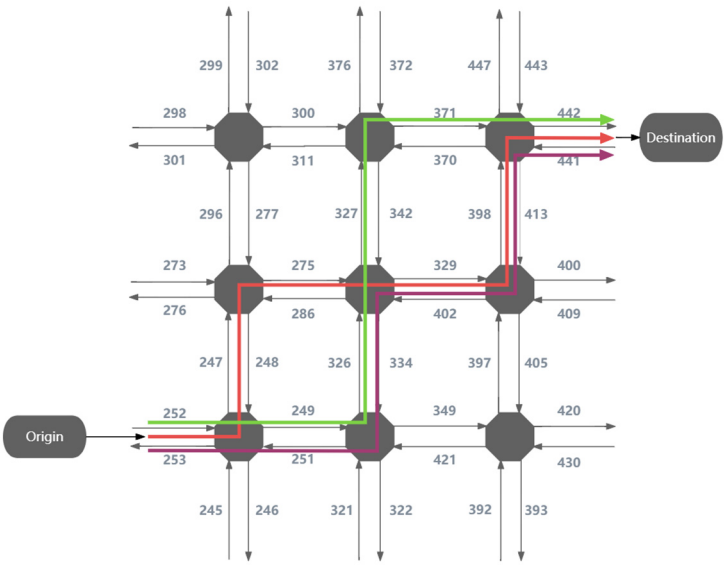

Figure 4. Simulated transport network in an AIMSUN environment. The red line represents the trajectory associated with OriginID = 1, the purple color represents the trajectory associated with OriginID = 2, and the green color represents the trajectory associated with OriginID = 3.

This section provides the training and testing outcomes of SCC-MIE and baseline models for AIMSUN. The performance review encompasses several facets and diverse performance criteria. In order to guarantee the ability to replicate the results, the specific parameters utilized throughout the training procedure are outlined in Table 3.

Table 3. Hyperparameters used for test.

Hyperparameter	Value
Number of iterations	10,000
Number of episodes	10,000
$\text{Batch}_{\text{size}}$	2048
Number of hidden neurons	64
Learning rate	0.00003
Discount rate of reward	0.99
Entropy coefficient	0.01

6.1. Baseline

- **Mobility Markov Chain [26]**: Mobile Markov chain (MMC) is a fundamental model commonly used to address location prediction problems. MMC models represent a stochastic process where transitions occur between different states in the state space. One key characteristic of MMC models is their "memoryless" nature, meaning that the probability distribution of transitioning to the next state is solely determined by the current state.
- **Recurrent Neural Network [27]**: Recurrent neural network (RNN) is a popular and robust algorithm for location recommendation tasks. RNN models excel at capturing the spatiotemporal characteristics in data, allowing them to make accurate predictions for the next location. In this study, we employ long short-term memory (LSTM) for modeling continuous data.
- **Inverse Reinforcement Learning [28]**: Inverse reinforcement learning (IRL) aims to infer unknown reward functions based on observed demonstrations or expert behaviors in order to train RL agents or guide their decision-making process in new, unknown environments. In this study, we employ maximum entropy IRL (MaxEnt) to extend the idea of matching state visits to matching state-action visits.

- **Generative Adversarial Imitation Learning [6]**: GAIL is an imitation learning algorithm based on generative adversarial networks (GAN), which allows a learner to learn strategies to imitate experts by confronting them.
- **Deconfounded Multi-agent Environment Reconstruction [17]**: DEMER uses a multi-agent generative adversarial imitation learning framework. It is proposed to introduce a confounder embedding strategy and train the strategy using a compatible discriminator.

6.2. Result

This section presents the training and testing results of the model. It begins with the convergence curves for the training results, followed by the accuracy of trajectory similarity. Next, it compares the accuracy of expert and learner models based on imitation learning. Finally, it compares the running time of each model.

Convergence Curve: Figure 5 illustrates the convergence curve of the loss functions (Discrim_loss,Policy_loss,Value_loss) and the causal entropy $(\mathrm{H}(\pi_\theta))$ based on the AIMSUN dataset. From Figure 5, we find that when the number of iterations is small, both discrim_loss and Value_loss decrease, while policy_loss increases. For example, when the number of iterations starts, Discrim_loss is 1.38, Policy_loss is -1.44, and Value_loss is 0.49. When the number of iterations is increased to 20, the changes in each loss function are more obvious, i.e., Discrim_loss shows a steep drop to 0.53, while Policy_loss decreases to -0.93, and Value_loss decreases to 0.14. As the number of iterations is 30, Discrim_loss is 0.12, Policy_loss is -0.14, Value_loss is 0.0015. As the number of iterations is 40, Discrim_loss is 0.08, Policy_loss is -0.11, Value_loss is 0.00011. This is because the policy generator and the discriminator have different design purposes in the framework. The main goal of the policy generator is to learn to generate action sequences that are similar to the expert's policy by maximizing Policy_loss in order to achieve high-quality trajectory generation. In contrast, the discriminator is designed to minimize the gap between the action sequences generated by the generator and the real samples so that it can accurately distinguish between the two. In addition, the framework introduces the concept of confidence level so that the SCC-MIE discriminator can guide the training process of the generator more effectively. When the generator returns a high confidence level, the discriminator has a lower probability of misclassifying the generated action sequences. Thus, the loss function of the discriminator decreases rapidly, which motivates the generator to generate trajectories closer to the expert's strategy. This mechanism accelerates the training process and improves the efficiency of the algorithm.

However, as training continues, the discriminator begins to discriminate real trajectories from the generated trajectories to the point where the accuracy of the generator continues to decrease. Therefore, in the middle of iterations, discrim_loss tends to increase, and policy_loss tends to decrease. For example, when the number of iterations is 200, we can observe that discrim_loss = 1.07, policy_loss = -0.59. As the number of iterations increases to 500, we find that discrim_loss is 1.32 and policy_loss is -0.63. When the number of iterations varies to 10,00, we find that discrim_loss is 1.38 and policy_loss is -0.66.

In addition, we find that value_loss is decreasing and almost converges to zero for the increasing number of iterations. For example, when the number of iterations is 1000, value_loss = 0.00054. At this point, the entropy H_{π_θ} converges to the maximum value of causal entropy. To ensure that the results are correct, we increased the number of training times to 10,000 and realized that none of the values had changed significantly. This is because as the number of iterations increases, all the values reach convergence to the point at which the discriminator is unable to distinguish the real trajectory from the generated trajectory. Therefore, SCC-MIE achieves the best trajectory generation through environment reconstruction.

Accuracy of Trajectory Similarity: Figure 6 depicts the precision of the trajectories produced by each model over the five route models. We employed two widely utilized assessment metrics in sequence modelling to evaluate the similarity at the trajectory level:

the BLEU [29] score and the METEOR [30] score. Both ratings have a maximum value of 1, while higher values indicate superior accuracy.

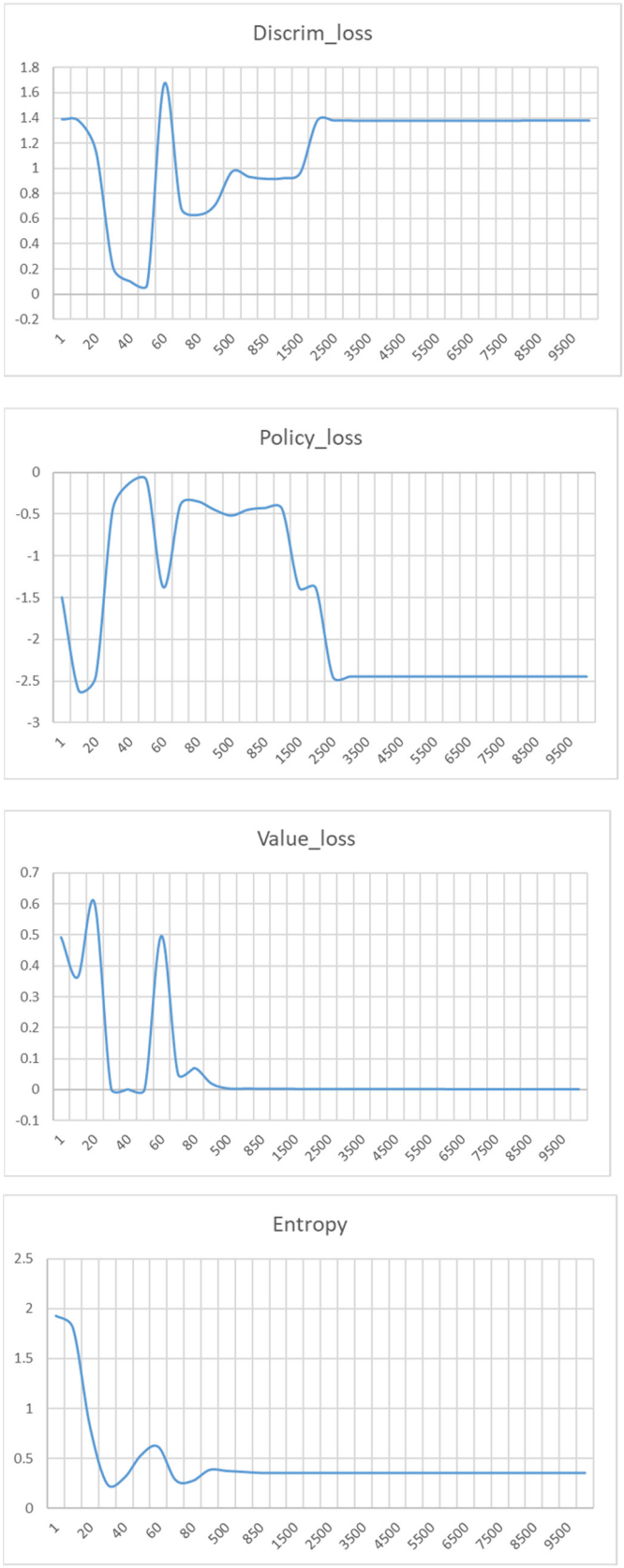

Figure 5. Loss functions of Discrim, Policy, Value, and Entropy.

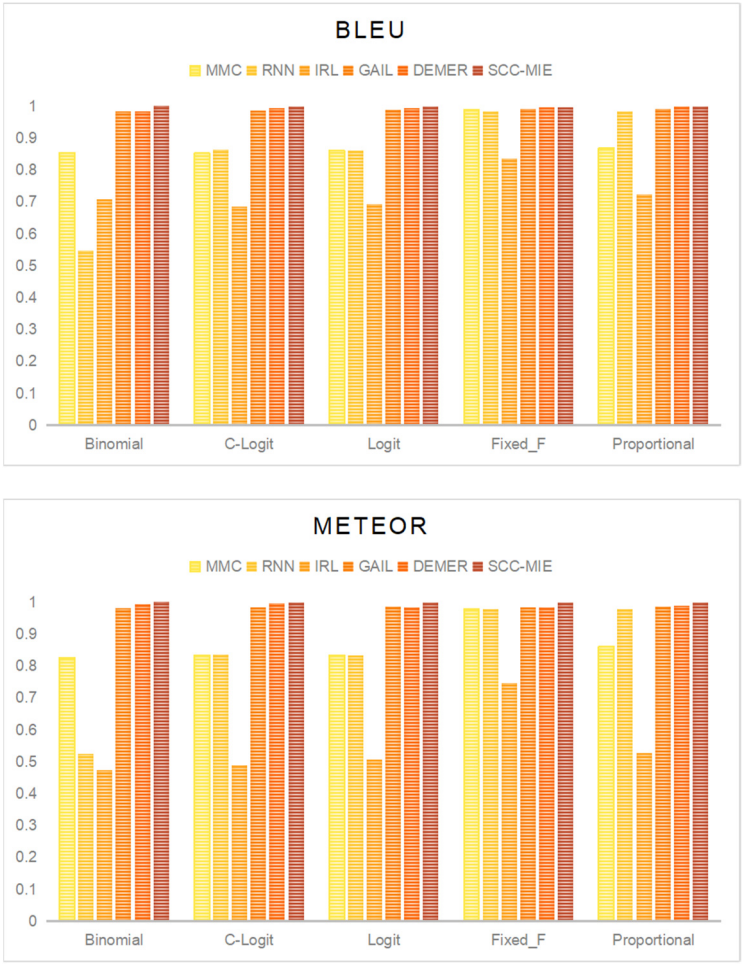

Figure 6. Accuracy of trajectory similarity.

The findings indicate that DEMER and SCC-MIE exhibit considerably higher levels of accuracy across all five route models. As an illustration, DEMER achieves a mean BLEU score of 0.9931 and a mean METEOR score of 0.9887, whereas SCC-MIE attains a mean BLEU score of 0.9986 and a mean METEOR score of 0.9985. Conversely, the IRL method has the poorest performance overall, as seen by its lowest BLEU score of 0.6843 in the C-logit model and its lowest METEOR score of 0.4724 in the binomial model.

Accuracy of Expert and Learner: Figure 7 exhibits that the accuracy of the expert and the learner is different for the number of training iterations. In this experiment, we divide the entire dataset into a training dataset and a test dataset in the ratio of 0.7:0.3. The number of training iterations is dynamically varied in the range [10, 10,000]. Then, all models use 1000 sample trajectories to compare the model's performance properly. Table 3 provides further information regarding the hyperparameters of SCC-MIE. The learner's accuracy is shown to improve with an increasing number of training rounds, as depicted in Figure 7. For example, the number of iterations is 10, the accuracy of the expert is 100%, and the accuracy of the learner is 51.54%. When the number of iterations is increased to 100, the accuracy of the expert is 85.68%, whereas the accuracy of the learner is 84.43%. The accuracy of novice learners is expected to be poorer due to their incomplete understanding of the expert's method.

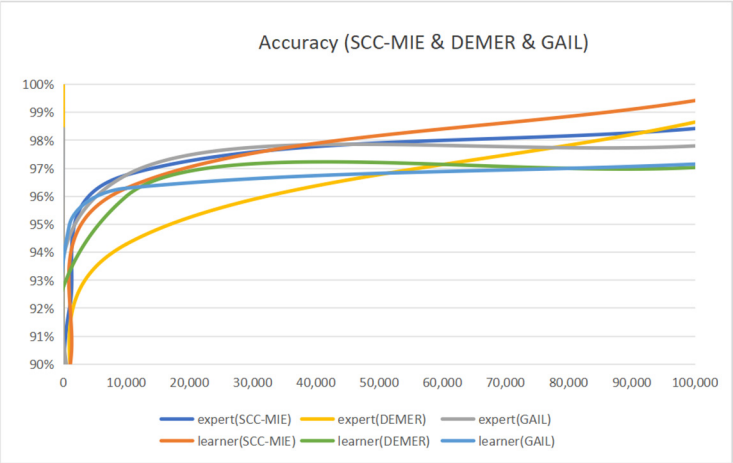

Figure 7. Accuracy of SCC-MIE, DEMER, and GAIL.

In addition, the learner's accuracy shows large fluctuations because he/she is still trying to understand and imitate the expert's strategy. As the training progresses, the learner gradually improves his/her strategy, and the accuracy increases. More often than not, the learner will gradually converge to a strategy that approaches or exceeds the expert's, thus stabilizing accuracy and eventually meeting or exceeding it. For instance, the number of iterations is 10,000, the accuracy of the expert is 96.76%, and the learner's accuracy is 96.29%. The number of iterations is 100,000, the accuracy of the expert is 98.42%, and the learner's accuracy is 99.42%.

Therefore, the model training relies on expert experience at first. Then, as the number of training sessions increases, the learner gains experience, and the accuracy rate continues to rise.

Moreover, we can see that the accuracy of learners in GAIL is highly variable from Figure 7. For example, when the number of iterations is 10, the expert achieves a perfect accuracy of 100%, whereas the learner's accuracy is 54.92%. When the number of iterations increases to 100, the expert's accuracy is 85.75%, while the learner's accuracy is 86.26%. Due to the limitations of real environments in providing complete and observable information, the reconstruction of the environment is hindered.

DEMER attempts to reconstruct hidden confounding factors to better structure the environment. The curve of the learner in DEMER follows a similar trend to the curve in the other models. For example, when the number of iterations is 10, the expert reaches a perfect accuracy of 100% and the learner's accuracy is 69.7%. When the number of iterations is increased to 100, the expert's accuracy is 85.71% and the learner's accuracy is 71.88%. The DEMER model provided inspiration in providing complete and observable information, so we developed a new approach called SCC-MIE. This approach improves the performance of GAIL (generative adversarial imitation learning) and DEMER (deconfounded multi-agent environment reconstruction) and demonstrates the beneficial effects of our confounding factor setting.

Running time: Figure 8 depicts the computation time, measured on a GPU, required to generate 10,000 vehicle trajectories using six different models. Interestingly, it is observed that all six models exhibit the capability to generate the desired 10,000 vehicle trajectories in less than 2 s. Figure 9 (SCC-MIE) demonstrates that the running times of the six approaches are relatively similar when the number of iterations reaches 1000.

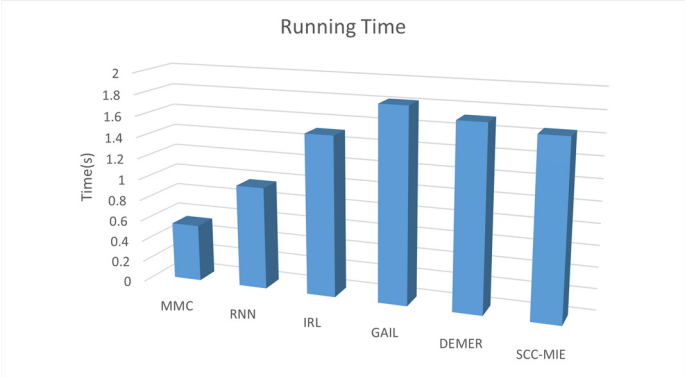

Figure 8. Running time taken to generate 10,000 vehicle trajectories.

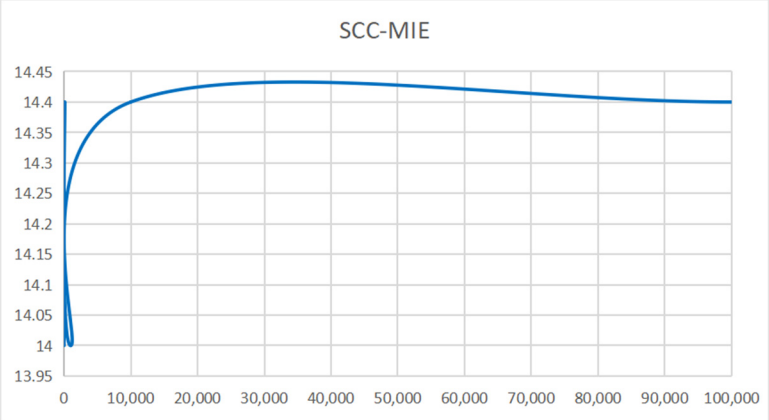

Figure 9. Running time of SCC-MIE.

In general, when the number of iterations is comparatively lower, the above trajectory generation algorithm may have a relatively shorter running time. However, virtual environments in large trajectory recommendation tasks are highly dynamic and hybrid, requiring a large number of iterations to achieve good results. When the number of iterations keeps increasing, the running time of algorithms, such as MMC, IRL, GAIL, DEMER, etc., may show a nonlinear increase because of the complexity of the environment construction and the high dimensionality of the state space. In contrast, the increase in SCC-MIE is relatively small and the application value is high.

7. Conclusions

In this paper, we proposed self-interested coalitional crowdsensing for multi-agent interactive environment monitoring (SCC-MIE), based on the generative adversarial training framework to construct a virtual sensing environment with hidden confounders to conduct the environment monitoring in real-time. To begin with, we incorporated the confounder-embedded policy into the generator and ensured compatibility of the discriminator with various classification tasks, thus facilitating precise optimization of each strategy. After that, we introduced a novel self-interested coalitional learning scheme that can cooperate with additional discriminators to provide interpretable confidence. The confidence level obtained from this module can be utilized for worker selection, ensuring that the chosen workers contribute more meaningful data for reconstructing the sensing environment. Finally, we evaluated the framework using an AIMSUN-based trajectory dataset. Extensive experiments against state-of-the-art baselines have demonstrated the effectiveness and robustness of our approach.

Author Contributions: Methodology, X.L. (Xiuwen Liu), X.L. (Xinghua Lei); Software, S.C.; Formal analysis, X.L. (Xiuwen Liu); Investigation, X.L. (Xiuwen Liu); Writing—original draft, X.L. (Xinghua Lei); Writing—review & editing, X.L. (Xiuwen Liu), S.C.; Supervision, X.L. (Xiuwen Liu), X.L. (Xin Li); Funding acquisition, X.L. (Xiuwen Liu), X.L. (Xin Li). All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported in part by the Natural Science Foundation of Shandong Province of China (ZR202103040180).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ganti, R.K.; Ye, F.; Lei, H. Mobile crowdsensing: Current state and future challenges. *IEEE Commun. Mag. Artic. News Events Interest Commun. Eng.* **2011**, *49*, 32–39. [CrossRef]
2. Capponi, A.; Fiandrino, C.; Kantarci, B.; Foschini, L.; Bouvry, P. A Survey on Mobile Crowdsensing Systems: Challenges, Solutions and Opportunities. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 2419–2465. [CrossRef]
3. Ye, Z.; Xiao, K.; Ge, Y.; Deng, Y. Applying Simulated Annealing and Parallel Computing to the Mobile Sequential Recommendation. *IEEE Trans. Knowl. Data Eng.* **2019**, *31*, 243–256. [CrossRef]
4. Zhang, C.; Zhu, L.; Xu, C.; Ni, J.; Huang, C.; Shen, X. Location privacy-preserving task recommendation with geometric range query in mobile crowdsensing. *IEEE Trans. Mob. Comput.* **2021**, *21*, 4410–4425. [CrossRef]
5. Canese, L.; Cardarilli, G.C.; Di Nunzio, L.; Fazzolari, R.; Giardino, D.; Re, M.; Spanò, S. Multi-agent reinforcement learning: A review of challenges and applications. *Appl. Sci.* **2021**, *11*, 4948. [CrossRef]
6. Ho, J.; Ermon, S. Generative Adversarial Imitation Learning. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 4565–4573.
7. Foerster, J. Deep Multi-Agent Reinforcement Learning. Ph.D. Thesis, University of Oxford, Oxford, UK, 2018.
8. Li, T.; Zhu, K.; Luong, N.C.; Niyato, D.; Wu, Q.; Zhang, Y.; Chen, B. Applications of multi-agent reinforcement learning in future internet: A comprehensive survey. *IEEE Commun. Surv. Tutor.* **2022**, *24*, 1240–1279. [CrossRef]
9. Schmidt, L.M.; Brosig, J.; Plinge, A.; Eskofier, B.M. An introduction to multi-agent reinforcement learning and review of its application to autonomous mobility. In Proceedings of the 2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC), Macau, China, 8–12 October 2022; pp. 1342–1349.
10. Xie, K.; Tian, J.; Xie, G.; Zhang, G.; Zhang, D. Deep learning-enabled sparse industrial crowdsensing and prediction. In Proceedings of the IEEE INFOCOM 2021-IEEE Conference on Computer Communications, Vancouver, BC, Canada, 10–13 May 2021; pp. 1–10.
11. Wang, E.; Zhang, M.; Cheng, X.; Yang, Y.; Liu, W.; Yu, H.; Wang, L.; Zhang, J. Low cost sparse network monitoring based on block matrix completion. In Proceedings of the IEEE Transactions on Industrial Informatics, Vancouver, BC, Canada, 10–13 May 2021; pp. 6170–6181.
12. Shi, J.C.; Yu, Y.; Da, Q.; Chen, S.Y.; Zeng, A.X. Virtual-taobao: Virtualizing real-world online retail environment for reinforcement learning. *Proc. AAAI Conf. Artif. Intell.* **2019**, *33*, 4902–4909. [CrossRef]
13. Liu, C.; Wang, L.; Wen, X.; Liu, L.; Zheng, W.; Lu, Z. Efficient Data Collection Scheme based on Information Entropy for Vehicular Crowdsensing. In Proceedings of the 2022 IEEE International Conference on Communications Workshops (ICC Workshops), Seoul, Republic of Korea, 16–20 May 2022; pp. 1–6.
14. Qin, H.; Zhan, X.; Li, Y.; Yang, X.; Zheng, Y. Network-wide traffic states imputation using self-interested coalitional learning. In Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery Data Mining, Virtual Event, 14–18 August 2021; pp. 1370–1378.
15. Liu, W.; Wang, E.; Yang, Y.; Wu, J. Worker selection towards data completion for online sparse crowdsensing. In Proceedings of the IEEE INFOCOM 2022-IEEE Conference on Computer Communications, Virtual Event, 2–5 May 2022; pp. 1509–1518.
16. Wu, A.; Luo, W.; Yang, A.; Zhang, Y.; Zhu, J. Efficient Bilateral Privacy-Preserving Data Collection for Mobile Crowdsensing. *IEEE Trans. Serv. Comput.* **2023**. [CrossRef]
17. Shang, W.; Yu, Y.; Li, Q.; Qin, Z.; Meng, Y.; Ye, J. Environment reconstruction with hidden confounders for reinforcement learning based recommendation. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery Data Mining, Anchorage, AK, USA, 4–8 August 2019; pp. 566–576.
18. Ma, J.; Guo, R.; Chen, C.; Zhang, A.; Li, J. Deconfounding with networked observational data in a dynamic environment. In Proceedings of the 14th ACM International Conference on Web Search and Data Mining, Virtual Event, 8–12 March 2021; pp. 166–174.
19. Zhao, Y.; Liu, C.H. Social-aware incentive mechanism for vehicular crowdsensing by deep reinforcement learning. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 2314–2325. [CrossRef]

20. Dai, Z.; Liu, C.H.; Ye, Y.; Han, R.; Yuan, Y.; Wang, G.; Tang, J. Aoi-minimal uav crowdsensing by model-based graph convolutional reinforcement learning. In Proceedings of the IEEE INFOCOM 2022-IEEE Conference on Computer Communications, Virtual Event, 2–5 May 2022; pp. 1029–1038.
21. Schaal, S. Is imitation learning the route to humanoid robots? *Trends Cogn. Sci.* **1999**, *3*, 233–242. [CrossRef] [PubMed]
22. Le Mero, L.; Yi, D.; Dianati, M.; Mouzakitis, A. A survey on imitation learning techniques for end-to-end autonomous vehicles. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 14128–14147. [CrossRef]
23. Song, J.; Ren, H.; Sadigh, D.; Ermon, S. Multi-agent generative adversarial imitation learning. *Adv. Neural Inf. Process. Syst.* **2018**, *31*. [CrossRef]
24. Nemhauser, G.L.; Wolsey, L.A.; Fisher, M.L. An analysis of approximations for maximizing submodular set functionsi. *Math. Program.* **1978**, *14*, 265–294. [CrossRef]
25. Choi, S.; Kim, J.; Yeo, H. TrajGAIL: Generating urban vehicle trajectories using generative adversarial imitation learning. *Trans-Portation Res. Part C Emerg. Technol.* **2021**, *128*, 103091. [CrossRef]
26. Gambs, S.; Killijian, M.O.; del Prado Cortez, M.N. Next place prediction using mobility markov chains. In Proceedings of the First Workshop on Measurement, Privacy, and Mobility, Bern, Switzerland, 10 April 2012; pp. 1–6.
27. Altaf, B.; Yu, L.; Zhang, X. Spatio-temporal attention based recurrent neural network for next location prediction. In Proceedings of the 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 10–13 December 2018; pp. 937–942.
28. Pieter, A.; Ng, A.Y. Apprenticeship learning via inverse reinforcement learning. In Proceedings of the Twenty-First International Conference on Machine Learning (ICML '04), Banff, AB, Canada, 4–8 July 2004; Association for Computing Machinery: New York, NY, USA, 2004. [CrossRef]
29. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.J. BLEU: A method for automatic evaluation of machine translation. In Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, Association for Computational Linguistics, Philadelphia, PA, USA, 6–12 July 2002; pp. 311–318.
30. Banerjee, S.; Lavie, A. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, Ann Arbor, MI, USA, 29–30 June 2005; pp. 65–72.