CF-GODE: Continuous-Time Causal Inference for Multi-Agent Dynamical Systems

Song Jiang
songjiang@cs.ucla.edu
University of California, Los Angeles
Los Angeles, CA

Zijie Huang
zijiehuang@cs.ucla.edu
University of California, Los Angeles
Los Angeles, CA

Xiao Luo
xiaoluo@cs.ucla.edu
University of California, Los Angeles
Los Angeles, CA

Yizhou Sun
yzsun@cs.ucla.edu
University of California, Los Angeles
Los Angeles, CA

ABSTRACT

Multi-agent dynamical systems refer to scenarios where multiple units (aka agents) interact with each other and evolve collectively over time. For instance, people's health conditions are mutually influenced. Receiving vaccinations not only strengthens the long-term health status of one unit but also provides protection for those in their immediate surroundings. To make informed decisions in multi-agent dynamical systems, such as determining the optimal vaccine distribution plan, it is essential for decision-makers to *estimate the continuous-time counterfactual outcomes*. However, existing studies of causal inference over time rely on the assumption that units are mutually independent, which is not valid for multi-agent dynamical systems. In this paper, we aim to bridge this gap and study how to estimate counterfactual outcomes in multi-agent dynamical systems. Causal inference in a multi-agent dynamical system has unique challenges: 1) Confounders are time-varying and are present in both individual unit covariates and those of other units; 2) Units are affected by not only their own but also others' treatments; 3) The treatments are naturally dynamic, such as receiving vaccines and boosters in a seasonal manner. To this end, we model a multi-agent dynamical system as a graph and propose a novel model called CF-GODE (**C**ounter**F**actual **G**raph **O**rdinary **D**ifferential **E**quations). CF-GODE is a causal model that estimates continuous-time counterfactual outcomes in the presence of inter-dependencies between units. To facilitate continuous-time estimation, we propose Treatment-Induced GraphODE, a novel ordinary differential equation based on graph neural networks (GNNs), which can incorporate dynamical treatments as additional inputs to predict potential outcomes over time. To remove confounding bias, we propose two domain adversarial learning based objectives that learn balanced continuous representation trajectories, which are not predictive of treatments and interference. We further provide theoretical justification to prove their effectiveness. Experiments on two semi-synthetic datasets confirm that CF-GODE outperforms baselines on counterfactual estimation. We also provide extensive analyses to understand how our model works.

CCS CONCEPTS

• **Mathematics of computing → Causal networks**.

KEYWORDS

Causal Inference, Multi-Agent Dynamical System, GraphODE

ACM Reference Format:
Song Jiang, Zijie Huang, Xiao Luo, and Yizhou Sun. 2023. CF-GODE: Continuous-Time Causal Inference for Multi-Agent Dynamical Systems. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '23), August 6–10, 2023, Long Beach, CA, USA.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3580305.3599272

1 INTRODUCTION

Estimating counterfactual outcomes *over time* is critical to gaining causal understanding for many useful practical applications, such as how to distribute the limited vaccines in the early days to maximize protection over time [32], or how to design proper scheduling of medical treatments to optimize the patient recovery process [3]. Randomized controlled trials (RCTs) are the gold standard for causal inference, but they can be cost-prohibitive and ethically challenging, particularly when considering the dynamical settings described above. Therefore, estimating counterfactual outcomes from observational data is the key approach to answering causal questions in real-world scenarios. Existing research on observational causal inference over time has begun by utilizing basic linear regression [40] and Gaussian processes [54] to capture the time-dependencies. Subsequently, advancements have been made by incorporating more advanced deep learning models such as recurrent neural networks (RNNs) [3, 11] and Transformers [33].

Despite the progress, all aforementioned studies have relied on the assumption that units (e.g., people in the vaccine example) are independent of each other, i.e., each unit is solely influenced by its own treatment but not by others. In many realistic scenarios, however, this assumption is not valid. For instance, a person's vaccination not only protects themselves but also those close to them. This type of setting is referred to as a *multi-agent dynamical system* [13], where units (also known as agents) interact with each other and evolve collectively over time. Many practical problems can be expressed as multi-agent dynamical systems, such as

KDD '23, August 6–10, 2023, Long Beach, CA, USA

ACM ISBN 979-8-4007-0103-0/23/08.
https://doi.org/10.1145/3580305.3599272

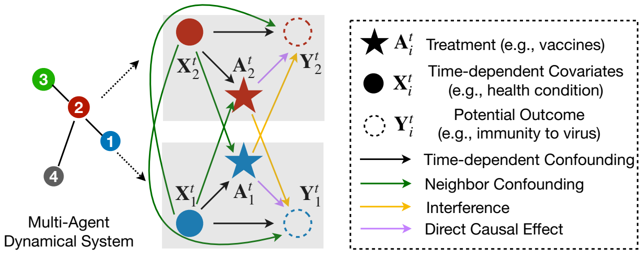
3
2
1
4
Multi-Agent Dynamical System
X_2^t
A_2^t
Y_2^t
X_1^t
A_1^t
Y_1^t
A_i^t Treatment (e.g., vaccines)
X_i^t Time-dependent Covariates (e.g., health condition)
Y_i^t Potential Outcome (e.g., immunity to virus)
Time-dependent Confounding
Neighbor Confounding
Interference
Direct Causal Effect

Figure 1: Causal graph at time t in a multi-agent dynamical system. The causal variables are represented by shapes, while their relationships are distinguished by colors.

the long-term effects of vaccination where people mutually influence [18], brain network signals in which the regions of interest (ROI) in a brain are associated [7, 57], and molecular systems movements where the atoms are interconnected [9]. Prior approaches for causal inference over time are not applicable to multi-agent dynamical systems since they are not capable of handling the interconnections between units. In this paper, we propose to study this novel problem *counterfactual estimation in multi-agent dynamical systems*, which has received limited attention in the literature.

The dynamic and interrelated nature of multi-agent dynamical systems poses unique and nontrivial challenges to causal inference. We illustrate them along with Fig. 1. 1) **Multi-source confounders**. Confounders are variables that have an impact on both treatments and outcomes, leading to spurious correlations between them. Therefore, in observational data, the treatments are not balanced among units with different confounders values, resulting in biased counterfactual outcomes estimation. For example, old people are more likely to receive vaccines, but also face a higher risk of virus infection. If we train a standard supervised model using such imbalanced data, it may wrongly predict that vaccines may increase the infection risk for young people. In multi-agent dynamical systems, the confounders are multi-source, including *time-dependent confounders* and *neighbor confounders*. Time-dependent confounders refer to the fact that the confounders typically evolve over time and thus their impact on treatments and outcomes also changes dynamically [3, 35]. For instance, people's health conditions change over time, affecting their likelihood of getting vaccinated and future health status. Neighbor confounders mean that a unit's treatment and outcome could also be confounded by the covariates of its near units (neighbors) [1, 10], For example, if family members are in poor health, a unit may be more likely to receive a vaccine. Compared to the independent setting, neighbor confounders are additional confounding factors in multi-agent dynamical systems. 2) **Imbalance of interference**. As discussed in the previous example that vaccines protect not only a unit but also those in close proximity, the outcome of a unit can be influenced by others' treatments in multi-agent dynamical systems. In causal language, this phenomenon is referred to as *interference*. Similar to the treatments, interference is affected by the covariates and thus is not balanced across the units in observational data [10]. For instance, highly educated units are more likely to receive vaccines and typically have more highly educated friends. Therefore, they receive stronger protection through higher vaccination rates among their social networks. Such imbalanced interference causes additional bias in the estimation of counterfactual outcomes. 3) **Continuous dynamics**. In realistic applications, a multi-agent dynamical system is continuous in nature [37]. However, most existing causal models are discrete, making them inappropriate for multi-agent dynamical systems. Modeling continuous-time observations (such as covariates and outcomes) and continuously estimating counterfactual outcomes over time remains an open challenge.

In this paper, we address the above challenges and study how to estimate continuous-time counterfactual outcomes, in presence of multi-source confounders and interference, in multi-agent dynamical systems. This is a novel, yet challenging and under-explored problem with valuable real-world applications.

To this end, we model a multi-agent dynamical system as a *graph*, where nodes represent units and edges capture their interactions. Inspired by recent achievements in graph ordinary differential equations (GraphODE) [19], we propose CF-GODE, a novel *causal* model that estimates continuous-time **C**ounter**F**actual outcomes based on **G**raph **O**rdinary **D**ifferential **E**quations in multi-agent dynamical systems. Specifically, we use GraphODE as a backbone to model the continuous trajectory of each unit. However, in this case, traditional GraphODE can only model the pure dynamics of potential outcomes [19] and lacks the ability to incorporate additional inputs such as treatments, making it inappropriate for causal inference. To address this issue, in CF-GODE, we propose *Treatment-Induced GraphODE*, a new GraphODE model capable of handling treatments when predicting the future trajectory of potential outcomes. Treatment-Induced GraphODE uses graph neural networks (GNNs) [26] to formulate its differential equations, which can effectively capture the mutual dependencies between units including neighbor confounders and interference. This advantage makes it a natural fit for counterfactual estimation in multi-agent dynamical systems. Then a latent representation is learned for each unit from its observations as the solution to Treatment-Induced GraphODE, which represents the continuous trajectory driven by treatments. The core of ensuring CF-GODE is a causal model is to deal with the aforementioned estimation bias caused by imbalanced treatments and interference in the observational data. We solve this issue via domain adversarial learning [12, 52], in which we treat the values of treatments (and interference) as domains and ensure the latent representation trajectories are invariant to them. We provide theoretical justification to demonstrate that the domain-adversarial balancing objective functions proposed in CF-GODE can effectively achieve the balancing goal, thereby removing bias in counterfactual estimation and ensuring that CF-GODE is causal.

We summarize our major **contributions** as follows: 1) We study how to estimate counterfactual outcomes in multi-agent dynamical systems, which is a novel yet challenging problem with useful practical implications. 2) We propose CF-GODE, a novel causal model for causal inference multi-agent dynamical systems based on GraphODE and domain-adversarial learning. 3) We provide theoretical analysis to show that CF-GODE is able to handle the imbalanced treatments and interference, ensuring unbiased counterfactual estimation. 4)We conduct extensive experiments to evaluate CF-GODE's performance on counterfactual outcomes estimation in multi-agent dynamical systems.

2 RELATED WORK

2.1 Causal Inference Over Time

The central challenge in estimating counterfactual outcomes in longitudinal settings is to remove the confounding bias from time-dependent covariates [3, 35]. The core solution to this in existing works is to cut off the association between covariates and the observed treatment assignments over time. To achieve this goal, statistical tools are widely used in traditional approaches. For example, marginal structural models (MSMs) [40] use inverse probability of treatment weighting (IPTW) [41, 42] to balance the distribution of covariates over time between unit groups that are assigned to different treatments. By doing so, treatment assignment can no longer be predicted from the balanced covariates, thus breaking their correlation. A later work [27] further enhances MSMs by using recurrent neural networks (RNNs) to learn the inverse probability of treatment weights (IPTWs), which is more capable of modeling sequential data. However, IPTW based tools can result in high variances in practice [3]. To overcome this limitation, recent studies [3, 33] extend the representation learning based balancing approaches from static settings [22, 47, 55, 56] to dynamic settings. Specifically, counterfactual recurrent network (CRN) [3] uses RNNs to encode the time-varying covariates into latent embeddings over time, which are simultaneously optimized by two objectives: potential outcomes prediction and longitudinal distribution balancing w.r.t. treatment assignments. The learned balanced embeddings are not predictive of treatments, thus ensuring unbiased estimates of the potential outcomes. Since RNNs are less powerful in capturing long-range dependencies, [33] improves CRN by using Transformer [49] that preserves long-range dependencies between time-dependent confounders. Despite the progress, all the above models can only predict counterfactual outcomes in discrete timestamps. However, practical longitudinal sequences are continuous in nature.

Our work is most related to [2, 8, 17, 46], which estimate counterfactual outcomes in continuous dynamic settings using neural ordinary differential equations (ODEs) [5, 43] or neural controlled differential equations (CDEs) [24]. Specifically, [46] infers continuous latent trajectories to represent the movement of potential outcomes and balance the distribution of this latent representation between treated and control groups via adversarial learning. However, [46] (and all aforementioned models) assume that units are mutually independent, which is usually not valid in many practical scenarios where the units affect each other, e.g., getting vaccinated provides long-term protection not only for oneself but also for their close ones. In contrast, our model is designed to estimate counterfactual outcomes in longitudinal settings where units are interdependent, i.e., the multi-agent dynamical systems.

2.2 Continuous Modeling With Neural Ordinary Differential Equations (ODEs)

Many dynamical systems are continuous in nature, which can be typically modeled using first-order ordinary differential equations (ODEs) [37]. ODEs describe a system's rate of change over time by a specific function, which is traditionally designed by domain experts [38], and more recently parameterized by neural networks [5, 31, 43], as a closed-form ODE function may be unknown for some complex real-world systems. Given initial states, the solution of a NeuralODE can be easily computed using any ODE solver, such as the Runge-Kutta method [45]. In multi-agent dynamical systems, such as the spread of infectious disease among people, units often interact with one another, yet standard NeuralODEs do not explicitly model these interactions. Recent works have sought to address this limitation by representing the interactions among multiple units as graphs, and then utilizing graph neural networks (GNNs) [26, 51] to parameterize the ODE function [19, 20, 36, 58]. When predicting the dynamics of each unit, these GraphODE models not only take into account the unit's own latent state but also aggregate the latent states of its connected units along the interaction graph, to effectively capture the mutual influence between them. However, GraphODE models are standard statistical methods and therefore lack the capability for causal inference. Instead, our model aims to address the unique challenges present in multi-agent dynamical systems, i.e., time-dependent confounders and network interference, in order to make counterfactual predictions.

3 PROBLEM SETUP

3.1 Problem Formulation

We study how to estimate counterfactual outcomes in the context of multi-agent dynamical systems, where the units engage in mutual interactions and evolve simultaneously over time. Throughout this paper, we use boldface uppercase letters to denote matrices or vectors, boldface uppercase letters with subscripts to signify elements of matrices or vectors, regular lowercase letters to represent values of variables, and calligraphic uppercase letters to indicate sets. We summarize all notations used in this paper in Appendix. A.4.

Formally, a multi-agent dynamical system can be represented by a dynamical graph $\mathcal{G}^t = (\mathcal{V}, \mathcal{E}^t)$, where $\mathcal{V} = \{v_1, v_2, ..., v_N\}$ is the set of N units (nodes) and $\mathcal{E}^t$ denotes the edge set at time t. An edge in $\mathcal{E}^t$ describes the intersection between the two units it connects at time t. In this paper, we present an early exploration of causal inference in multi-agent dynamical systems, and for the purpose of simplicity, we assume that the graph structure remains constant over time, i.e., $\mathcal{G}^t = \mathcal{G}$. Each unit is associated with time-varying variables, which are the causal quantities in our case. We introduce them together with the causal framework in the following.

We follow the longitudinal potential outcomes framework [39, 44] to formalize the counterfactual outcome estimation as in [3, 46]. The observational data $\left((\mathbf{X}^t, \mathbf{A}^t, \mathbf{Y}^t) \cup \mathbf{V}\right)$ in a multi-agent dynamical system contains time-dependent covariates $\mathbf{X}^t$ (e.g., health condition), dynamical treatments $\mathbf{A}^t$ (e.g., vaccine allocation), and time-varying outcomes $\mathbf{Y}^t$ (e.g., immunity to infectious disease). It is worth noting that $\mathbf{Y}^t$ is essentially a part of $\mathbf{X}^t$. $\mathbf{V}$ denotes the static covariates of units such as ethnicity. Let the historical records of the multi-agent dynamical system up to time t be represented by $\mathcal{H}^t = \{\bar{\mathbf{X}}^t, \bar{\mathbf{A}}^t, \bar{\mathbf{Y}}^t, \mathbf{V}\}$, where $\bar{\mathbf{X}}^t, \bar{\mathbf{A}}^t, \bar{\mathbf{Y}}^t$ are all the $\mathbf{X}^{t^-}, \mathbf{A}^{t^-}, \mathbf{Y}^{t^-}$ until t $(t^- \leq t)$, respectively. In causal inference, we are focused on understanding the potential outcomes $\mathbf{Y}^{t^+}(\mathbf{A}^{t^+} = a)$[1] that may occur in the future $(t^+ > t)$ under a specific treatment a, which explains the impact of the treatment assignment on the dynamics of the system. Note that a is a treatment trajectory that includes

[1] The potential outcome can also be formalized using *do* operation [34].

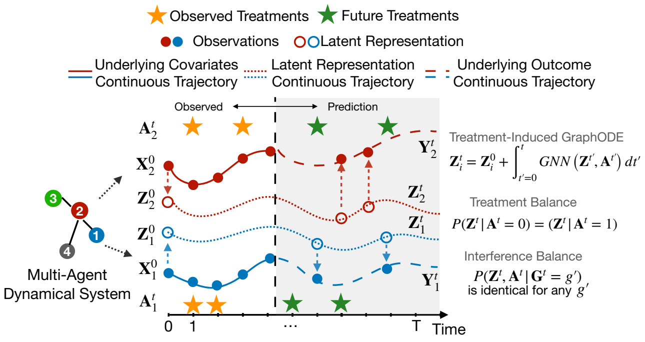
Observed Treatments
Future Treatments
Observations
Latent Representation
Underlying Covariates Continuous Trajectory
Latent Representation Continuous Trajectory
Underlying Outcome Continuous Trajectory
Observed
Prediction
$\mathbf{A}_2^t$
$\mathbf{X}_2^0$
$\mathbf{Z}_2^0$
$\mathbf{Z}_1^0$
$\mathbf{X}_1^0$
$\mathbf{A}_1^t$
$\mathbf{Y}_2^t$
$\mathbf{Z}_2^t$
$\mathbf{Z}_1^t$
$\mathbf{Y}_1^t$
3
2
1
4
Multi-Agent Dynamical System
Treatment-Induced GraphODE
$\mathbf{Z}_i^t = \mathbf{Z}_i^0 + \int_{t'=0}^{t} GNN\left(\mathbf{Z}^{t'}, \mathbf{A}^{t'}\right) dt'$
Treatment Balance
$P(\mathbf{Z}^t \mid \mathbf{A}^t = 0) = (\mathbf{Z}^t \mid \mathbf{A}^t = 1)$
Interference Balance
$P(\mathbf{Z}^t, \mathbf{A}^t \mid \mathbf{G}^t = g')$ is identical for any g'
0
1
...
T
Time

Figure 2: Overview of CF-GODE. **The initial latent representation Z^0 is first learned from initial observations. Then the continuous latent representation trajectory Z^t is learned as the solution to treatment-induced GraphODE, which is able to handle treatments as additional inputs. The graph neural network (GNN) based ODE function naturally models the mutual dependencies. The future potential outcomes can be decoded from Z^t at any given time. To remove confounding bias, Z^t is balanced with respect to 1) treatments, 2) interference when combined with corresponding treatments.**

all treatments in the future time. Our goal is to estimate the future potential outcomes sequence driven by treatments in a multi-agent dynamical system, which is formalized as:

$$\mathbb{E}\left(\mathrm{Y}^{t^+}(\mathrm{A}^{t^+} = a) \mid \mathcal{H}^t, \mathcal{G}\right). \tag{1}$$

3.2 Causal Identification

The potential outcomes represented by $\mathrm{Y}^{t^+}(\mathrm{A}^{t^+} = a)$ are a causal quantity. To make it identifiable from observational data, we must adhere to the following necessary assumptions.

Assumption 1: Positivity (Overlap). The future treatment trajectory is probabilistic regardless of the historical observation, i.e., $0 < P(\mathrm{A}^{t^+} = a \mid \mathcal{H}^t) < 1, \forall \mathcal{H}^t$.

Assumption 2: Consistency. Under the same treatment trajectory a, the potential outcome is equal to the observed outcomes, i.e., $\mathrm{Y}^{\mathrm{t}^+}(\mathrm{A}^{t^+} = a) = Y^{t^+}$.

The above two assumptions are standard for longitudinal counterfactual estimation. To identify the potential outcomes, it is also necessary to assume that there are no unobserved confounders, i.e., the strong ignorability assumption. However, the typical sequential strong ignorability assumption [3, 27, 33, 46] is not appropriate for multi-agent dynamical systems, because the graph structure $\mathcal{G}$ introduces extra graph confounders and interference. A plausible strong ignorability assumption for graphs is first introduced by [10] and later validated in studies such as [21, 29, 30]. However, the assumption made in these works is limited to static settings. To address this, we extend it to longitudinal settings and adapt it to be applicable to multi-agent dynamical systems in the following.

We first introduce a summary function, denoted as $g(\cdot)$, that captures the interference effects caused by the treatments of a node's neighboring units in the graph as in [10]. Formally, $\mathrm{G}_i^t = g(\mathrm{A}_{\mathcal{N}_i}^t, \mathrm{A}_{\mathcal{N}_{-i}}^t)$, where $\mathrm{A}_{\mathcal{N}_i}^t$ denotes the treatments of node i's immediate neighbors, and $\mathrm{A}_{\mathcal{N}_{-i}}^t$ is the treatments of all the remaining node that are not directly connected to node i. We refer to G_i^t as *interference summary*. Here for simplicity, we adopt the assumption put forth in [1, 10, 21] that a node is only influenced by the treatments of its immediate neighbors, i.e., $g(\mathrm{A}_{\mathcal{N}_i}^t, \mathrm{A}_{\mathcal{N}_{-i}}^t) = g(\mathrm{A}_{\mathcal{N}_i}^t, \mathrm{A'}_{\mathcal{N}_{-i}}^t) = g(\mathrm{A}_{\mathcal{N}_i}^t), \forall \mathrm{A}_{\mathcal{N}_{-i}}^t, \mathrm{A'}_{\mathcal{N}_{-i}}^t$. $g(\cdot)$ can be instantiated using any aggregation functions or models. As in previous studies [10, 21, 30], in this paper, we define G_i^t as the proportion of treated units in unit i's neighbors, i.e., $\mathrm{G}_i^t := \sum_{j \in \mathcal{N}_i} \frac{\mathrm{A}_i^t}{|\mathcal{N}_i|}$. With G_i^t, we present the strong ignorability assumption for multi-agent dynamical systems in the following:

Assumption 3: Strong Ignorability for Multi-Agent Dynamical Systems[2]. Given the historical observations and the graph structure that describes the multi-agent dynamical system, the potential outcome trajectory is independent of the treatments and interference summary, i.e., $\mathrm{Y}^{t^+}(\mathrm{A}^{t^+} = a) \perp\!\!\!\perp \mathrm{A}^{t^+}, \mathrm{G}^{t^+} \mid \mathcal{H}^t, \mathcal{G}. \forall a, t$.

With these three assumptions, the potential outcome trajectory Eq. (1) can be identifiable as:

$$\begin{aligned} &\mathbb{E}\left(\mathrm{Y}^{t^+}(\mathrm{A}^{t^+} = a) \mid \mathcal{H}^t, \mathcal{G}\right) \\ &= \mathbb{E}\left(\mathrm{Y}^{t^+}(\mathrm{A}^{t^+} = a) \mid \mathrm{A}^{t^+}, \mathrm{G}^{t^+}, \mathcal{H}^t, \mathcal{G}\right) && (2) \\ &= \mathbb{E}\left(\mathrm{Y}^{t^+} \mid \mathrm{A}^{t^+}, \mathrm{G}^{t^+}, \mathcal{H}^t, \mathcal{G}\right). && (3) \end{aligned}$$

Eq. (2) is true because of assumption 3, while Eq. (3) holds under the assumption 2. The above causal identification enables us to estimate the potential outcomes in multi-agent dynamical systems using observational data. More specifically, we can train a machine learning model on observational data, which takes treatment trajectory A^{t^+}, interference summary G^{t^+}, historical observation $\mathcal{H}^t$ and graph $\mathcal{G}$ as inputs, and the observed (factual) outcome Y^{t^+} as targets, to predict the counterfactual outcomes given new treatment trajectories. Our proposed model CF-GODE is grounded in this and will be presented in detail in the subsequent section.

4 PROPOSED MODEL: CF-GODE

4.1 Overview

Our proposed CF-GODE is a causal model that predicts counterfactual outcomes in a multi-agent dynamical system by learning from observational data. We show an overview of our model in Fig. 2. Compared to most existing causal models designed for standard sequential settings that consider discrete time intervals and independent units [3, 33], multi-agent dynamical systems are more realistic and present two challenging properties: the dynamics are *continuous* in nature, and units are *influenced* by others. To address these, our proposed CF-GODE takes the advantage of recent breakthroughs in graph ordinary differential equations (GraphODE) [19, 20] and extends it to handle treatments and interference, enabling continuous estimation of counterfactual outcomes in multi-agent dynamical systems. We refer to our ODE model as *Treatment-Induced GraphODE* (Sec. 4.2). The time-dependent confounders lead the distribution of covariates to be quite discrepant between units assigned to different treatments, resulting in high variances in counterfactual outcome estimation [22, 40, 47]. This effect is further amplified by the imbalanced interference caused by the graph

[2] Note that similar to the strong ignorability assumptions in static or non-graph sequential settings, assumption 3 can not be verified only from data.

structure in multi-agent dynamical systems [10, 21]. CF-GODE uses adversarial learning to alleviate this issue and guarantee unbiased estimates of counterfactual outcomes (Sec. 4.3).

4.2 Treatment-Induced GraphODE

To facilitate continuous-time counterfactual outcome estimation, we propose to learn a continuous latent trajectory $\mathbf{Z}_i^t$ for every node in multi-agent dynamical system that represents their movement. An ideal $\mathbf{Z}_i^t$ should possess two characteristics: 1) the ability to predict observed outcomes, and 2) not to be predictive of the received treatment or interference in observational data[3]. We implement such a $\mathbf{Z}_i^t$ by a novel model called Treatment-Induced GraphODE, which empowers the recent GraphODE [19, 20] to deal with treatment and interference for counterfactual outcomes estimation.

In a multi-agent dynamical system, the future outcomes of node i might be affected by not only its own past movement and current treatment, but also the movements and interference from neighbors (e.g., a unit's health condition and vaccination status have a significant impact on how likely others are to be infected). We model this process and formalize treatment-induced GraphODE as:

$$\mathbf{Z}_i^t = \mathbf{Z}_i^0 + \int_{t'=0}^{t} \phi\left(\mathbf{Z}^{t'}, \mathbf{A}^{t'}\right) dt'. \tag{4}$$

In Eq. (4), $\mathbf{Z}^{t'}$ and $\mathbf{A}^{t'}$ denote the latent trajectory representations and treatments of all nodes in the multi-agent dynamical system, respectively. $\phi(\cdot)$ is the ODE function. To comprehensively capture the effects from node i and its connected neighbors, we parameterize $\phi(\cdot)$ using graph neural networks [26] with self-loops. $\mathbf{Z}_i^0$ is the initial state and can be encoded from the initial observations as $\mathbf{Z}_i^0 = f(\mathbf{X}_i^0, \mathbf{V}_i)$, where $f(\cdot)$ is an encoder parameterized by neural networks. With $\mathbf{Z}_i^0$, we can obtain the $\mathbf{Z}_i^t$, which is the solution to treatment-induced GraphODE, by solving an ODE initial-value problem (IVP) in Eq. (4), formalized as:

$$\mathbf{Z}_i^0, \mathbf{Z}_i^1 \cdots \mathbf{Z}_i^T = \text{ODESolve}\left(\phi, [\mathbf{Z}_1^0, \mathbf{Z}_2^0 \cdots \mathbf{Z}_N^0], (t_0, t_1 \cdots t_T)\right), \tag{5}$$

where T is the number of timestamps for the evaluation of Eq. (5). With the solution latent trajectory $\mathbf{Z}_i^t$, we can then use a decoder $d_{\mathbf{Y}}(\cdot)$ to transform it to the predicted outcome $\hat{\mathbf{Y}}_i^t = d_{\mathbf{Y}}(\mathbf{Z}_i^t)$. We also use neural networks to instantiate $d_{\mathbf{Y}}(\cdot)$. We compare the prediction $\hat{\mathbf{Y}}_i^t$ to ground-truths $\mathbf{Y}_i^t$ in all observed timestamps $(t_0, t_1 \cdots t_T)$ using a mean square error as objective, which is formalized as:

$$L^{\langle Y \rangle} = \frac{1}{N}\frac{1}{T}\sum_i^N \sum_t^T \left(\hat{\mathbf{Y}}_i^t - \mathbf{Y}_i^t\right)^2. \tag{6}$$

4.3 Balancing via Adversarial Learning

In the observational data, the treatments applied to each unit $\mathbf{A}_i^t$ are affected by the time-dependent confounders present in the covariates (and thus in its latent representation trajectory $\mathbf{Z}_i^t$). Consequently, the distribution of latent representation trajectory is not balanced among units with different treatment assignments, i.e., $P(\mathbf{A}_i^t|\mathbf{Z}_i^t)$ is not uniform, leading to high variances in the counterfactual outcome estimation [22, 47]. In the context of multi-agent dynamical systems, this effect is further exacerbated by the presence of *imbalanced interference* among units. This is because a unit's interference is influenced by its covariates (also the latent representation) and treatments in the observational data, i.e., $P(\mathbf{G}_i^t|\mathbf{Z}_i^t, \mathbf{A}_i^t)$ is not uniform [10, 21, 30]. Here we give an intuitive example of the imbalanced interference: consider that a highly educated person is more likely to be surrounded by other highly educated friends, who believe in science and are more likely to be vaccinated, thereby providing stronger protection for this person against infectious diseases, i.e., higher interference.

A sufficient condition to remove the above bias is to ensure that the distribution of latent representation trajectories is invariant to treatments, and when combined with the corresponding treatments, is interference-invariant [3, 10, 21, 46, 47]. This condition is formalized as $P(\mathbf{Z}^t|\mathbf{A}^t = 0) = (\mathbf{Z}^t|\mathbf{A}^t = 1)$ for treatment balancing, and $P(\mathbf{Z}^t, \mathbf{A}^t|\mathbf{G}^t = g')$ is identical for any given value of g' for interference balancing. The treatment $\mathbf{A}^t$ is binary and interference $\mathbf{G}^t$ is continuous as in [10, 21]. Note that the aforementioned conditions are over the unit groups. This guarantees that the treatment cannot be inferred from the latent representation trajectory, and that the interference is not predictable when the treatment is combined with latent representation. We implement this balancing goal through domain adversarial learning [12], in which the treatment is treated as binary domains and the interference is treated as continuous domains [52]. Specifically, we use the gradient reversal layer proposed in [12], denoted as $r(\cdot)$, to adversarially optimize the latent representation trajectory at every observed time, making it agnostic towards the treatments and interference.

Treatment Balancing. Formally, the predicted treatment is $\hat{\mathbf{A}}_i^t = d_{\mathbf{A}}\left(r(\mathbf{Z}_i^t)\right)$, where the $d_{\mathbf{A}}$ is a neural network that attempts to recover the treatment from latent representation. The gradient reversal layer $r(\cdot)$ does nothing in the forward pass, but reverses the gradients in the back-propagation. This way, a min-max game is created in which $d_{\mathbf{A}}$ aims to minimize the treatment prediction loss, while the latent representation learner in treatment-induced GraphODE strives to maximize it, as formalized in the following:

$$L^{\langle A \rangle} = \min_{d_{\mathbf{A}}^j} \max_{f,\phi} \frac{1}{N}\frac{1}{T}\sum_i^N \sum_t^T \sum_{j\in\{0,1\}} \mathbb{1}_{(\mathbf{A}_i^t = j)} - \log\left(d_{\mathbf{A}}^j\left(r(\mathbf{Z}_i^t)\right)\right), \tag{7}$$

where $d_{\mathbf{A}}^j$ represents the logits of $d_{\mathbf{A}}(\cdot)$ for predicting treatment j. We then provide a theoretical analysis to justify the capability of $L^{\langle A \rangle}$ to attain balanced representations in the following.

Theorem 1. *Let $j \in \{0, 1\}$ be the binary treatment values, and let N and T denote the number of units and observed timestamp lengths, respectively. Let $P_j^t = P(\mathbf{Z}^t \mid \mathbf{A}^t = j)$, be the distribution of latent representation $\mathbf{Z}^t$ for the group of units with treatments j at time t. Let f, ϕ, $d_{\mathbf{A}}^j$ be the initial state encoder, the ODE function of treatment-induced GraphODE, and logits of predicting treatment j. The necessary and sufficient condition for the min-max game in Eq. (7) to be optimal is $P_0^t = P_1^t, \forall t \in (t_0, t_1 \cdots t_T)$.*

Theorem 1 suggests that the condition to obtain global optimum of Eq. (7) is $P(\mathbf{Z}^t|\mathbf{A}^t = 0) = (\mathbf{Z}^t|\mathbf{A}^t = 1)$. Therefore, by optimizing $L^{\langle A \rangle}$ in Eq. (7), we can ensure the latent representation trajectory $\mathbf{Z}^t$ is balanced with respect to treatments. In other words, $\mathbf{Z}^t$ is not predictive of $\mathbf{A}^t$. We prove Theorem 1 in Appendix. A.1.1.

[3] The second characteristic is discussed in Sec. 4.3.

Interference Balancing. The interference $\mathbf{G}_i^t$ is continuous, we thus adapt the continuous domain adversarial learning [52] to achieve the interference balancing. Similar to the binary case, we consider the continuous interference as continuous domains, and use the gradient reversal layer $r(\cdot)$ to build a min-max game on interference prediction as follows:

$$L^{\langle G \rangle} = \min_{d_{\mathrm{G}}} \max_{f,\phi} \frac{1}{N}\frac{1}{T}\sum_{i}^{N}\sum_{t}^{T}\left(d_{\mathrm{G}}\left(r([\mathbf{Z}_i^t, \mathbf{A}_i^t])\right) - \mathbf{G}_i^t\right)^2 \tag{8}$$

where d_{G} is the interference predictor which is parameterized by neural networks, and $[\cdot,\cdot]$ is the concatenation operation. In the following, we also theoretically demonstrate that $L^{\langle G \rangle}$ is able to achieve the interference balancing objective.

Theorem 2. *Let f, ϕ, d_{G} be the initial state encoder, the ODE function of treatment-induced GraphODE, and the interference predictor. The necessary and sufficient condition for min-max game in Eq. (8) to be optimal is $P(\mathbf{Z}^t, \mathbf{A}^t | \mathbf{G}^t = g')$ is identical for any g'.*

Theorem. 2 indicates that if $\mathbb{E}([\mathbf{Z}^t, \mathbf{A}^t] \mid \mathbf{G}^t)$ is identical for any $\mathbf{G}^t = g'$, Eq. (8) achieves optimum. Therefore, it is sufficient to balance the combination of representations and treatments with respect to interference $\mathbf{G}^t$ by optimizing the objective function $L^{\langle G \rangle}$. We show the proof of Theorem 2 in Appendix. A.1.2.

4.4 Training of CF-GODE

Objective Function. The overall objective function of CF-GODE is formalized in the following:

$$L = L^{\langle Y \rangle} + \alpha_{\mathrm{A}} L^{\langle A \rangle} + \alpha_{\mathrm{G}} L^{\langle G \rangle}, \tag{9}$$

where coefficients α_{A}, α_{G} are the strengths of the treatment balancing and interference balancing, respectively. By adversarially optimizing L, the latent representation trajectory $\mathbf{Z}_i^t$ is able to predict the outcome trajectory $\mathbf{Y}_i^t$ while remaining invariant to the treatments $\mathbf{A}_i^t$ and interference $\mathbf{G}_i^t$ (combined with treatments), which enables the unbiased counterfactual outcome estimation in multi-agent dynamical systems.

Alternative Training as Trade-Off. In practice, we find that directly training CF-GODE with the overall loss function L may not be stable as $L^{\langle A \rangle}$ and $L^{\langle G \rangle}$ could hinder the ability of latent representation trajectory $\mathbf{Z}_i^t$ to predict the outcome. Therefore, we trade-off the training of CF-GODE in an alternative manner between L and $L^{\langle Y \rangle}$, to ensure that $\mathbf{Z}_i^t$ is capable of predicting outcomes. Specifically, we switch the training iterations between L and $L^{\langle Y \rangle}$ with a ratio of K, i.e., $\frac{Iter_L}{Iter_{L^{\langle Y \rangle}}} = K$, where $Iter$ means the number of training iterations and K is a tunable hyperparameter. We elaborate on the training procedure in Appendix. A.2.

5 EXPERIMENTS

5.1 Experimental Settings

Dataset. In observational data, we only have factual outcomes but not counterfactual outcomes. Therefore, we use semi-synthetics data to evaluate CF-GODE as in [16, 29, 50]. That is, we use two real graphs Flickr and BlogCatalog [6, 16, 28] and use a Pharmacokinetic-Pharmacodynamic (PK-PD) model [15] to simulate the continuous trajectory of treatments and potential outcomes [3, 46]. The data simulation mimics the vaccine example in the real world. We introduce the data simulation process in detail in Appendix. A.3.

Metric. We focus on counterfactual outcomes estimation in this paper, which is a continuous value. Therefore, we use mean square errors (MSE) as our metric to evaluate the performance of our model, which is formalized as $MSE := \frac{1}{N}\frac{1}{T}\sum_i^N \sum_t^T \left(\hat{\mathbf{Y}}_i^t - \mathbf{Y}_i^t\right)^2$.

Baselines. The scope of our model is in continuous-time causal inference, therefore we compare CF-GODE with the following baselines: **CDE** [24]: Ordinary differential equations with external inputs to adjust the continuous trajectory. **GraphODE** [19] Ordinary differential equations model with graph neural networks (GNNs) based ODE functions. **TE-CDE** [46]: the state-of-the-art model for continuous-time counterfactual outcomes estimation based on neural controlled differential equations (NeuralCDE).

Implementation. The parameters of CF-GODE are set as follows: the dimension of latent representations is 64; the ODE solver is the Euler method; the balancing degrees are $\alpha_{\mathrm{A}} = \alpha_{\mathrm{G}} = 0.5$. For training hyperparameters, the learning rate is 0.0001; the default alternative training ratio K is 4. We train the model 5000 epochs and select the best model according to the performance on the validation set. The parameters are optimized by Adam [25]. We run all experiments on a Lambda Labs instance with one A100 GPU.

Table 1: Counterfactual outcomes estimation errors on two datasets. "BC" is the abbreviation of the BlogCatalog dataset. The errors are broken down in x-step future estimation ($x \in [1, 2, 3, 4, 5]$). MSE errors are reported. The best results are in boldface and the second best results are underlined. CF-GODE**-N is the variant of our model without any balancing;** CF-GODE**-T means balance only w.r.t. treatments;** CF-GODE**-I denotes balance only w.r.t. interference.**

Dataset	Model	1-step	2-step	3-step	4-step	5-step	Overall
Flickr	CDE	0.134±0.015	0.164±0.017	0.198±0.021	0.237±0.023	0.281±0.026	0.203±0.205
	GraphODE	0.237±0.013	0.276±0.010	0.313±0.016	0.347±0.018	0.379±0.021	0.310±0.016
	TE-CDE	0.189±0.025	0.216±0.021	0.246±0.027	0.281±0.041	0.326±0.063	0.252±0.031
	CF-GODE-N	0.089±0.006	0.102±0.007	0.114±0.009	0.126±0.010	0.139±0.012	0.114±0.008
	CF-GODE-T	0.058±0.017	0.066±0.020	0.075±0.023	0.084±0.027	0.098±0.036	0.076±0.025
	CF-GODE-I	0.069±0.007	0.080±0.008	0.091±0.009	0.103±0.011	0.115±0.012	0.092±0.009
	CF-GODE	**0.056±0.009**	**0.060±0.009**	**0.067±0.009**	**0.070±0.010**	**0.077±0.012**	**0.065±0.010**
BC	CDE	0.255±0.120	0.324±0.178	0.407±0.263	0.515±0.383	0.640±0.549	0.427±0.296
	GraphODE	0.195±0.018	0.223±0.023	0.251±0.028	0.280±0.033	0.309±0.040	0.252±0.028
	TE-CDE	0.316±0.086	0.351±0.089	0.399±0.093	0.493±0.137	0.725±0.351	0.457±0.127
	CF-GODE-N	0.167±0.012	0.188±0.015	0.209±0.018	0.228±0.023	0.246±0.028	0.207±0.019
	CF-GODE-T	**0.139±0.015**	**0.154±0.019**	**0.172±0.025**	**0.189±0.027**	**0.202±0.031**	**0.171±0.023**
	CF-GODE-I	0.164±0.016	0.188±0.020	0.210±0.024	0.232±0.029	0.253±0.035	0.209±0.025
	CF-GODE	0.148±0.015	0.166±0.019	0.186±0.023	0.205±0.025	0.229±0.029	0.186±0.021

5.2 Can CF-GODE Deliver Accurate Estimations of Counterfactual Outcomes in Multi-Agent Dynamical Systems?

We compare CF-GODE to three lines of models: 1) Continuous-time dynamical prediction models CDE and GraphODE. Note that these baselines are not causal models since they only preserve the dynamical statistical associations. 2) Continuous-time causal inference model TE-CDE. But it is not capable of capturing the mutual dependencies between units in multi-agent dynamical systems. 3) Variants of CF-GODE. We consider three variants: CF-GODE-N means there is no any balancing ($\alpha_{\text{A}} = \alpha_{\text{G}} = 0$); CF-GODE-T denotes balancing only w.r.t. treatments ($\alpha_{\text{A}} = 1, \alpha_{\text{G}} = 0$); CF-GODE-I means balancing only w.r.t. interference ($\alpha_{\text{A}} = 0$, $\alpha_{\text{G}} = 1$). For a multi-agent dynamical system with N nodes and length-A treatment trajectories, the total number of possible treatments for all nodes is $O(A \cdot 2^N)$. Therefore, it is intractable to enumerate all treatment combinations. To this end, we randomly flip 50% of all observed treatments in each experiment. We estimate five-step (timestamp) ahead counterfactual outcomes and report estimation errors in Table. 1. Generally, CF-GODE and the variants outperform the baselines by substantial margins. It is noteworthy that, despite being a causal model, TE-CDE performs clearly worse than the family of CF-GODE, because it ignores the mutual influence

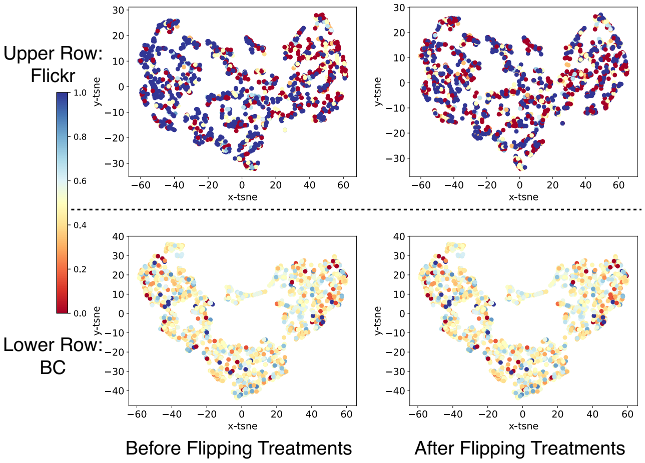

Figure 3: T-SNE projections of latent representations "before" (factual) and "after" (counterfactual) flipping the treatments. Each point represents a unit's latent representation. The points are colored by the units' corresponding interference. Upper row: Flickr dataset; Lower Row: BlogCatalog dataset.

Why Does CF-GODE**-T Show Superior Performance Than** CF-GODE **on BlogCatalog Dataset?** On BlogCatalog dataset, we observe that balancing solely with respect to treatments (CF-GODE-T) yields the lowest estimation errors, even outperforming balancing both treatments and interference (CF-GODE). To understand this phenomenon, we project all units' latent representations $\mathbf{Z}^t$ into 2-D embeddings using T-SNE [48] and color these 2-D points by their corresponding interference in Fig. 3. Specifically, we compare the units' interference before and after flipping the treatments. Compared to Flickr, we notice that in BlogCatalog 1) the latent representations are already comparatively more balanced before flipping the treatments, and 2) the units' interference does not change significantly after flipping the treatments. This suggests that balancing solely with respect to treatments might be sufficient in the BlogCatalog dataset. Actually, since we use the same data simulation protocol for Flickr and BlogCatalog, this difference in interference distribution is expected to be caused by their distinct graph structures. Specifically, the average and standard derivation of node degrees of the two datasets are Flickr: 2.0 ± 1.7; BlogCatalog: 30.7 ± 25.1. Intuitively, the interference of high degrees nodes is more resistant to flipping a random portion of their neighbors, which is pretty common among nodes in BlogCatalog. We provide further breakdown studies to better understand how node degrees affect counterfactual outcomes estimation in Sec. 5.5.

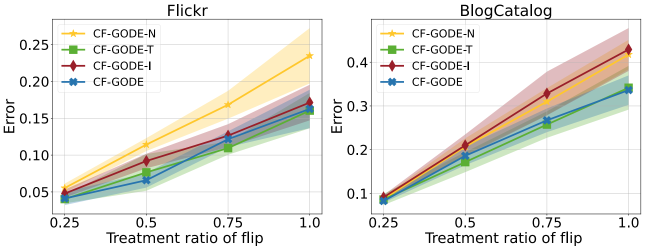

Figure 4: Counterfactual outcomes estimation errors w.r.t. the percentage of units in the graph whose treatments are flipped. Left: Flickr dataset; Right: BlogCatalog dataset.

5.3 How Does CF-GODE Respond to The Flipping of Counterfactual Treatments?

In the above experiments, the default treatment flipping ratio is set at 50%. It's intriguing to investigate how CF-GODE reacts to different flipping ratios, as this would indicate the degree of difference between factual and counterfactual outcomes in terms of treatments. To this end, we set the flip ratio as $[25\%, 50\%, 75\%, 100\%]$, and present the results of CF-GODE and its variants under these settings in Fig. 4. As expected, all models perform worse as the flip ratio increases, since the counterfactual treatments diverge further from the observed factual treatments. However, we observe that with balancing objectives, the error of CF-GODE increases generally slowly, highlighting the need for balancing objectives.

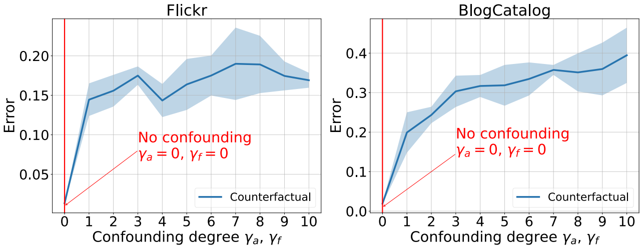

Figure 5: Counterfactual outcomes estimation errors w.r.t. 11 different confounding degrees γ_a and γ_f. Note γ_a and γ_f are set as the same values in each experiment. The red line points to the "no confounding bias" setting ($\gamma_a = \gamma_f = 0$).

5.4 How Does CF-GODE Respond to Different Confounding Degrees?

In data simulation, we use coefficients γ_a, γ_f to control the degree of the time-dependent and neighbor confounding bias. With larger values of these coefficients, the confounding bias is more severe, leading to increasingly imbalanced data. To study how CF-GODE works under varying confounding degrees, we set $\gamma_a = \gamma_f = \gamma$, where $\gamma \in [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]$, and present the counterfactual outcomes estimation errors of CF-GODE under these conditions in Fig.5. The errors increase as the confounding bias becomes more severe, but the rate of increase is relatively smooth, particularly on Flickr dataset. This implicates CF-GODE's robustness against high degree confounding bias. Additionally, CF-GODE produces low errors when there is no confounding bias ($\gamma_a = \gamma_f = 0$), which demonstrates the compatibility of CF-GODE with such settings.

5.5 How Does Graph Structure Impact Counterfactual Outcomes Estimation?

As discussed in Sec. 5.2, the graph structure affects CF-GODE's performance on counterfactual outcomes estimation. To gain deeper insights into this phenomenon, we break down the estimation errors on BlogCatalog dataset according to node degrees in Table. 2. Interestingly, we find that CF-GODE's counterfactual estimation errors decrease as the node degrees become higher. Intuitively, this might also be because the interference of high-degree nodes is more stable. However, in this paper, we do not have a theoretical understanding of the relationships between estimation errors and node degrees. We leave this line of research in future study.

Table 2: The breakdown of counterfactual outcomes estimation errors by units (nodes) degrees in BlogCatalog dataset.

Degree	#Nodes	Percentage%	Error
(0,5]	176	10.2	0.243±0.337
(5,10]	185	10.6	0.235±0.344
(10,20]	389	22.5	0.216±0.296
(20,30]	312	18.0	0.218±0.314
(20,30]	200	11.5	0.221±0.295
(30,40]	165	9.5	0.195±0.284
(40,50]	98	5.7	0.176±0.269
>50	207	12.0	0.187±0.261

5.6 Can CF-GODE Be Generalized to New Multi-Agent Dynamical Systems?

Standard counterfactual outcome estimations are typically conducted on units whose factual outcomes have been observed. However, the estimation of the potential outcomes on *new* multi-agent dynamical systems is also of great importance. For instance, to predict the effects of an initial vaccine distribution strategy for a new community. To assess CF-GODE's ability to generalize to new systems, i.e., new graphs, we split the original graph into three subgraphs, denoted as training/validation/testing graphs (details in Appendix. A.3). We train our model on the training graph and evaluate its potential outcome estimation on the testing graph. We report the results in Table. 3. We note that the performance of CF-GODE and the variants on new graphs are also generally better than baselines, which is consistent with the estimation of the counterfactual outcomes within the same graph (Sec. 5.2). This demonstrates our model's generalizability to new multi-agent dynamical systems.

Table 3: Generalization errors of potential outcomes prediction for new multi-agent dynamical systems (new graphs) on two datasets. "BC" is the abbreviation of the BlogCatalog dataset. The errors are broken down in x-step future estimation ($x \in [1, 2, 3, 4, 5]$). MSE errors are reported. The best results are in boldface and the second best results are underlined. CF-GODE**-N is the variant of our model without any balancing;** CF-GODE**-T means balance only w.r.t. treatments;** CF-GODE**-I denotes balance only w.r.t. interference.**

Dataset	Model	1-step	2-step	3-step	4-step	5-step	Overall
Flickr	CDE	0.134±0.017	0.166±0.022	0.201±0.026	0.241±030	0.285±0.034	0.205±0.025
	GraphODE	0.209±0.010	0.243±0.012	0.275±0.015	0.306±0.018	0.335±0.022	0.274±0.014
	TE-CDE	0.193±0.023	0.221±0.021	0.251±0.027	0.285±0.040	0.328±0.061	0.256±0.030
	CF-GODE-N	0.087±0.06	0.099±0.008	0.111±0.009	0.122±0.010	0.134±0.011	0.111±0.008
	CF-GODE-T	**0.057±0.014**	0.064±0.016	0.072±0.018	0.081±0.022	0.092±0.029	0.738±0.020
	CF-GODE-I	0.071±0.007	0.083±0.008	0.096±0.009	0.109±0.010	0.122±0.011	0.096±0.009
	CF-GODE	**0.057±0.008**	**0.062±0.009**	**0.067±0.010**	**0.073±0.011**	**0.081±0.013**	**0.069±0.010**
BC	CDE	0.251±0.113	0.317±0.174	0.399±0.259	0.500±0.380	0.625±0.548	0.418±0.294
	GraphODE	0.183±0.021	0.210±0.026	0.237±0.032	0.265±0.039	0.293±0.046	0.237±0.033
	TE-CDE	0.328±0.092	0.372±0.104	0.440±0.164	0.582±0.378	0.933±0.966	0.457±0.127
	CF-GODE-N	0.168±0.008	0.190±0.009	0.213±0.012	0.235±0.015	0.255±0.021	0.213±0.013
	CF-GODE-T	**0.141±0.011**	**0.157±0.015**	**0.175±0.021**	**0.192±0.022**	**0.203±0.028**	**0.174±0.018**
	CF-GODE-I	0.164±0.014	0.187±0.018	0.209±0.022	0.231±0.028	0.253±0.034	0.209±0.023
	CF-GODE	0.143±0.013	0.162±0.017	0.180±0.022	0.199±0.023	0.214±0.025	0.180±0.019

5.7 How Does Alternative Training Affect CF-GODE?

CF-GODE uses an alternative training strategy to trade off the latent representation balancing and potential outcome prediction. We examine how this trade-off is performed under varying alternative ratios K, where $K = \frac{Iter_L}{Iter_{L^{\langle Y \rangle}}}$ represents the alternating training between the overall loss L and the outcome prediction loss $L^{\langle Y \rangle}$. Fig. 6 shows the 2-D T-SNE projections of latent representations and their corresponding counterfactual estimation errors for different K values. We note that compared to solely training on $L^{\langle Y \rangle}$ (i.e., no balancing), training with L is able to force the embeddings more balanced, suggesting the effectiveness of our proposed domain adversarial learning based balancing objectives. In addition, with a smaller K, CF-GODE achieves better estimation errors, while a bigger K leads to more balanced latent representations. These results confirm that our alternative training is able to trade off between latent representation balancing and potential outcome prediction. In practice, choosing an appropriate value of K is expected to be determined through empirical analysis for each dataset.

5.8 Case Study: When CF-GODE Is Good, and When It Is Not.

To intuitively understand how CF-GODE works in estimating counterfactual outcomes and to study when CF-GODE would fail, we sample one successful unit and one failure unit from Flickr dataset. We draw their factual outcomes, counterfactual outcomes, and the estimations made by CF-GODE and CF-GODE-N (without balancing) in Fig. 7. In the successful case, the estimate by CF-GODE is able to conform to the counterfactual treatment trajectory, while

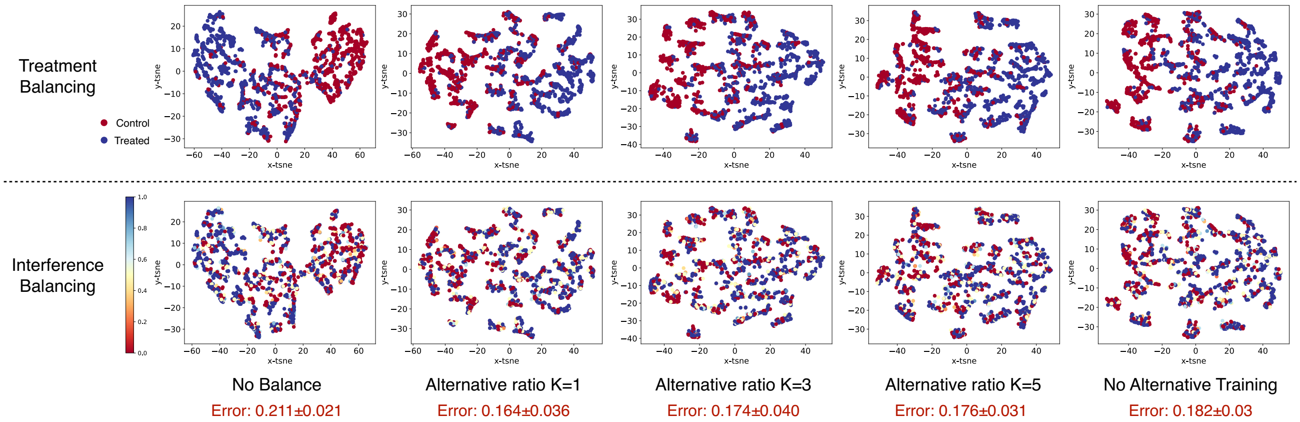

Figure 6: The T-SNE visualization of latent representations under different alternative training settings. Each point represents
ents and the bottom line is colored by observed
ing); 2) $Iter_L$ = 1; 3) $Iter_L$ = 3; 4) $Iter_L$ = 5;

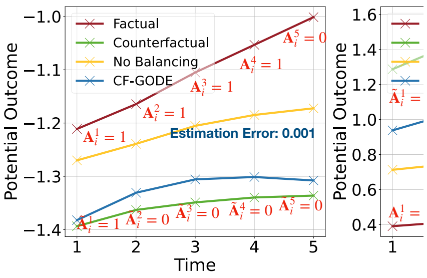

Figure 7: The predicted counterfactual o
w/w.o balancing loss functions. The tr
outcomes A_i^t and treatments of counter
are attached around the corresponding
tamp. The estimation errors are noted
from Flickr dataset. Left: a successful ca

CF-GODE-N still follows the factual trajector
tiveness of our proposed balancing objectives. However, CF-GODE also makes mistakes. In the failure case, its estimate fails to catch up with the counterfactual outcome trajectory, yielding a non-trivial error. We speculate that this is because the counterfactual outcome of this unit is quite distinct from the factual one in terms of data scale, making counterfactual estimation more difficult.

5.9 How Hyperparamters Affect CF-GODE?

The two balancing objectives are core to making CF-GODE causal. Therefore, we finally study the impact of their degrees, represented by α_{A} and α_{G} in the loss function, on the model performance. We test α_{A} and α_{G} values evenly ranging from $[0, 0, 1.0]$, and present the counterfactual outcomes estimation errors for each combination of α_{A} and α_{G} in Fig. 8. Our results show that the errors are relatively higher when both α_{A} and α_{G} are in low values, i.e., light balancing. On the other hand, with larger values, the estimation errors generally become lower, but with high variance. This indicates

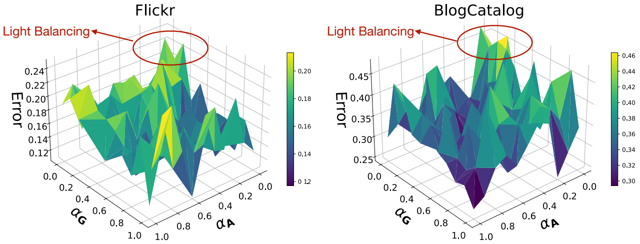

Figure 8: The counterfactual estimation errors w.r.t. different combinations of α_{A} and α_{G}. The "Light Balancing" settings where α_{A} and α_{G} are both in small values are circles in red.

6 CONCLUSION

In this paper, we study continuous-time counterfactual outcomes estimation in multi-agent dynamical systems, where units interact with each other. To this end, we propose CF-GODE, a novel causal model based on GraphODE to enable continuous potential outcomes prediction, and domain adversarial learning to remove confounding bias. We provide both theoretical justification and empirical analyses to demonstrate the effectiveness of our model. One limitation of CF-GODE is it needs the assumption of strong ignorability for multi-agent dynamical systems, which is not testable in practice. Recent studies relax this assumption by inferring latent proxy variables [53], which could be a potential solution.

ACKNOWLEDGMENTS

This work was partially supported by NSF 2211557, NSF 1937599, NSF 2119643, NSF 2303037, NASA, SRC, Okawa Foundation Grant, Amazon Research Awards, Cisco research grant, Picsart Gifts, and Snapchat Gifts.

REFERENCES

[1] David Arbour, Dan Garant, and David Jensen. 2016. Inferring network effects from observational data. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 715–724.

[2] Alexis Bellot and Mihaela Van Der Schaar. 2021. Policy analysis using synthetic controls in continuous-time. In *International Conference on Machine Learning*. PMLR, 759–768.

[3] Ioana Bica, Ahmed M. Alaa, James Jordon, and Mihaela van der Schaar. 2020. Estimating counterfactual treatment outcomes over time through adversarially balanced representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*.

[4] David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research* 3, Jan (2003), 993–1022.

[5] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. 2018. Neural ordinary differential equations. *Advances in neural information processing systems* 31 (2018).

[6] Zhixuan Chu, Stephen L Rathbun, and Sheng Li. 2021. Graph infomax adversarial learning for treatment effect estimation with networked observational data. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 176–184.

[7] Hejie Cui, Wei Dai, Yanqiao Zhu, Xuan Kan, Antonio Aodong Chen Gu, Joshua Lukemire, Liang Zhan, Lifang He, Ying Guo, and Carl Yang. 2022. BrainGB: a benchmark for brain network analysis with graph neural networks. *IEEE Transactions on Medical Imaging* (2022).

[8] Edward De Brouwer, Javier Gonzalez, and Stephanie Hyland. 2022. Predicting the impact of treatments over time with uncertainty aware neural differential equations.. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 4705–4722.

[9] Jacob D Durrant and J Andrew McCammon. 2011. Molecular dynamics simulations and drug discovery. *BMC biology* 9, 1 (2011), 1–9.

[10] Laura Forastiere, Edoardo M Airoldi, and Fabrizia Mealli. 2021. Identification and estimation of treatment and interference effects in observational studies on networks. *J. Amer. Statist. Assoc.* 116, 534 (2021), 901–918.

[11] Keisuke Fujii, Koh Takeuchi, Atsushi Kuribayashi, Naoya Takeishi, Yoshinobu Kawahara, and Kazuya Takeda. 2022. Estimating counterfactual treatment outcomes over time in multi-vehicle simulation. In *Proceedings of the 30th International Conference on Advances in Geographic Information Systems*. 1–4.

[12] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *The journal of machine learning research* 17, 1 (2016), 2096–2030.

[13] Veysel Gazi and Barış Fidan. 2007. Coordination and control of multi-agent dynamic systems: Models and approaches. In *Swarm Robotics: Second International Workshop, SAB 2006, Rome, Italy, September 30-October 1, 2006, Revised Selected Papers 2*. Springer, 71–102.

[14] Changran Geng, Harald Paganetti, and Clemens Grassberger. 2017. Prediction of treatment response for combined chemo-and radiation therapy for non-small cell lung cancer patients using a bio-mathematical model. *Scientific reports* 7, 1 (2017), 13542.

[15] Sylvain Goutelle, Michel Maurin, Florent Rougier, Xavier Barbaut, Laurent Bourguignon, Michel Ducher, and Pascal Maire. 2008. The Hill equation: a review of its capabilities in pharmacological modelling. *Fundamental & clinical pharmacology* 22, 6 (2008), 633–648.

[16] Ruocheng Guo, Jundong Li, and Huan Liu. 2020. Learning individual causal effects from networked observational data. In *Proceedings of the 13th International Conference on Web Search and Data Mining*. 232–240.

[17] Daehoon Gwak, Gyuhyeon Sim, Michael Poli, Stefano Massaroli, Jaegul Choo, and Edward Choi. 2020. Neural ordinary differential equations for intervention modeling. *arXiv preprint arXiv:2010.08304* (2020).

[18] M Elizabeth Halloran and Michael G Hudgens. 2012. Causal inference for vaccine effects on infectiousness. *The international journal of biostatistics* (2012).

[19] Zijie Huang, Yizhou Sun, and Wei Wang. 2020. Learning continuous system dynamics from irregularly-sampled partial observations. *Advances in Neural Information Processing Systems* 33 (2020), 16177–16187.

[20] Zijie Huang, Yizhou Sun, and Wei Wang. 2021. Coupled Graph ODE for Learning Interacting System Dynamics.. In *KDD*. 705–715.

[21] Song Jiang and Yizhou Sun. 2022. Estimating Causal Effects on Networked Observational Data via Representation Learning. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 852–861.

[22] Fredrik Johansson, Uri Shalit, and David Sontag. 2016. Learning representations for counterfactual inference. In *International conference on machine learning*. PMLR, 3020–3029.

[23] George Karypis and Vipin Kumar. 1998. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM Journal on scientific Computing* 20, 1 (1998), 359–392.

[24] Patrick Kidger, James Morrill, James Foster, and Terry Lyons. 2020. Neural controlled differential equations for irregular time series. *Advances in Neural Information Processing Systems* 33 (2020), 6696–6707.

[25] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

[26] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.

[27] Bryan Lim, Ahmed Alaa, and Mihaela Van Der Schaar. 2018. Forecasting treatment responses over time using recurrent marginal structural networks. *advances in neural information processing systems* 31 (2018).

[28] Jing Ma, Ruocheng Guo, Chen Chen, Aidong Zhang, and Jundong Li. 2021. Deconfounding with networked observational data in a dynamic environment. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. 166–174.

[29] Jing Ma, Mengting Wan, Longqi Yang, Jundong Li, Brent Hecht, and Jaime Teevan. 2022. Learning causal effects on hypergraphs. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1202–1212.

[30] Yunpu Ma and Volker Tresp. 2021. Causal inference under networked interference and intervention policy enhancement. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 3700–3708.

[31] Stefano Massaroli, Michael Poli, Jinkyoo Park, Atsushi Yamashita, and Hajime Asama. 2020. Dissecting neural odes. *Advances in Neural Information Processing Systems* 33 (2020), 3952–3963.

[32] Jan Medlock and Alison P Galvani. 2009. Optimizing influenza vaccine distribution. *Science* 325, 5948 (2009), 1705–1708.

[33] Valentyn Melnychuk, Dennis Frauen, and Stefan Feuerriegel. 2022. Causal Transformer for Estimating Counterfactual Outcomes. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA (Proceedings of Machine Learning Research, Vol. 162)*. PMLR, 15293–15329.

[34] Judea Pearl. 2009. *Causality*. Cambridge university press.

[35] Robert W Platt, Enrique F Schisterman, and Stephen R Cole. 2009. Time-modified confounding. *American journal of epidemiology* 170, 6 (2009), 687–694.

[36] Michael Poli, Stefano Massaroli, Junyoung Park, Atsushi Yamashita, Hajime Asama, and Jinkyoo Park. 2019. Graph neural ordinary differential equations. *arXiv preprint arXiv:1911.07532* (2019).

[37] Mason A Porter and James P Gleeson. 2014. Dynamical systems on networks: A tutorial. *arXiv preprint arXiv:1403.7663* (2014).

[38] Zhaozhi Qian, William Zame, Lucas Fleuren, Paul Elbers, and Mihaela van der Schaar. 2021. Integrating expert ODEs into Neural ODEs: Pharmacology and disease progression. *Advances in Neural Information Processing Systems* (2021).

[39] James M Robins and Miguel A Hernán. 2009. Estimation of the causal effects of time-varying exposures. *Longitudinal data analysis* 553 (2009), 599.

[40] James M Robins, Miguel Angel Hernan, and Babette Brumback. 2000. Marginal structural models and causal inference in epidemiology. , 550–560 pages.

[41] Paul R Rosenbaum. 1987. Model-based direct adjustment. *Journal of the American statistical Association* 82, 398 (1987), 387–394.

[42] Paul R Rosenbaum and Donald B Rubin. 1983. The central role of the propensity score in observational studies for causal effects. *Biometrika* 70, 1 (1983), 41–55.

[43] Yulia Rubanova, Tian Qi Chen, and David Duvenaud. 2019. Latent Ordinary Differential Equations for Irregularly-Sampled Time Series. In *Advances in Neural Information Processing Systems*.

[44] Donald B Rubin. 1978. Bayesian inference for causal effects: The role of randomization. *The Annals of statistics* (1978), 34–58.

[45] Michael Schober, Simo Särkkä, and Philipp Hennig. 2019. A probabilistic model for the numerical solution of initial value problems. *Statistics and Computing* 29, 1 (2019), 99–122.

[46] Nabeel Seedat, Fergus Imrie, Alexis Bellot, Zhaozhi Qian, and Mihaela van der Schaar. 2022. Continuous-Time Modeling of Counterfactual Outcomes Using Neural Controlled Differential Equations. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA (Proceedings of Machine Learning Research, Vol. 162)*. PMLR, 19497–19521.

[47] Uri Shalit, Fredrik D Johansson, and David Sontag. 2017. Estimating individual treatment effect: generalization bounds and algorithms. In *International Conference on Machine Learning*. PMLR, 3076–3085.

[48] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, 11 (2008).

[49] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[50] Victor Veitch, Yixin Wang, and David Blei. 2019. Using embeddings to correct for unobserved confounding in networks. *Advances in Neural Information Processing Systems* 32 (2019).

[51] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*.

[52] Hao Wang, Hao He, and Dina Katabi. 2020. Continuously Indexed Domain Adaptation. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event (Proceedings of Machine Learning Research, Vol. 119)*. PMLR, 9898–9907.

[53] Yixin Wang and David M Blei. 2019. The blessings of multiple causes. *J. Amer. Statist. Assoc.* 114, 528 (2019), 1574–1596.

[54] Yanbo Xu, Yanxun Xu, and Suchi Saria. 2016. A Bayesian nonparametric approach for estimating individualized treatment-response curves. In *Machine learning for healthcare conference*. PMLR, 282–300.

[55] Liuyi Yao, Sheng Li, Yaliang Li, Mengdi Huai, Jing Gao, and Aidong Zhang. 2018. Representation learning for treatment effect estimation from observational data. *Advances in Neural Information Processing Systems* 31 (2018).

[56] Jinsung Yoon, James Jordon, and Mihaela Van Der Schaar. 2018. GANITE: Estimation of individualized treatment effects using generative adversarial nets. In *International Conference on Learning Representations*.

[57] Yue Yu, Xuan Kan, Hejie Cui, Ran Xu, Yujia Zheng, Xiangchen Song, Yanqiao Zhu, Kun Zhang, Razieh Nabi, Ying Guo, et al. 2022. Learning Task-Aware Effective Brain Connectivity for fMRI Analysis with Graph Neural Networks. *arXiv preprint arXiv:2211.00261* (2022).

[58] Chengxi Zang and Fei Wang. 2020. Neural Dynamics on Complex Networks. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*. ACM, 892–902.

A APPENDIX

A.1 Proofs of Theorems

A.1.1 Proof of Theorem. 1 .

Theorem 1. *Let $j \in \{0,1\}$ be the binary treatment values, and let N and T denote the number of units and observed timestamp lengths, respectively. Let $P_j^t = P(\mathbf{Z}^t \mid \mathbf{A}^t = j)$, be the distribution of latent representation $\mathbf{Z}^t$ for the group of units with treatments j at time t. Let f, ϕ, $d_{\mathbf{A}}^j$ be the initial state encoder, the ODE function of treatment-induced GraphODE, and logits of predicting treatment j. The necessary and sufficient condition for the min-max game in Eq. (7) to be optimal is $P_0^t = P_1^t, \forall t \in (t_0, t_1 \cdots t_T)$.*

The proof of Theorem 1 follows [3, 33] and consists of two steps: to find the optimal $d_{\mathbf{A}}^j$ while fixing f and ϕ, and then to prove the optimal f and ϕ while fixing $d_{\mathbf{A}}^j$ can balance the latent representations, i.e., $P_0^t = P_1^t$. The first step is given by Proposition 1.

Proposition 1. *(Proposition 1 in [33]) Let $\alpha_j = P(A^t = j)$. When the initial state encoder f and ODE function ϕ are fixed, the optimal $d_{\mathbf{A}}^j$ at time t is:*

$$d_{\mathbf{A}}^{j\,*} = \frac{\alpha_j P_j^t}{\sum_{j' \in \{0,1\}} \alpha_{j'} P_{j'}^t}. \tag{10}$$

Proof. When fixing f and ϕ, $d_{\mathbf{A}}^{j\,*}$ is obtained by:

$$d_{\mathbf{A}}^{j\,*} = \underset{d_{\mathbf{A}}^j}{\operatorname{argmin}} \sum_{j \in \{0,1\}} \mathbb{1}_{(\mathbf{A}^t = j)} - \log\left(d_{\mathbf{A}}^j\left(r(\mathbf{Z}^t)\right)\right), \tag{11}$$

$$\text{subject to} \sum_{j \in \{0,1\}} d_{\mathbf{A}}^j\left(r(\mathbf{Z}^t)\right) = 1, \tag{12}$$

where Eq. (11) is adapted from Eq. (7). Note Eq. (11) can be applied to any i and t in Eq. (7), so we disregard the expectation with respect to them. Let $\alpha_j = P(A^t = j)$ and $P_j^t = P(\mathbf{Z}^t \mid \mathbf{A}^t = j)$. Then Eq. (11) can be rewritten as:

$$d_{\mathbf{A}}^{j\,*} = \underset{d_{\mathbf{A}}^j}{\operatorname{argmin}} \sum_{j \in \{0,1\}} -\mathbb{E}_{\mathbf{Z}^t \sim P_j^t} \alpha_j \log\left(d_{\mathbf{A}}^j\left(r(\mathbf{Z}^t)\right)\right) \tag{13}$$

$$= \underset{d_{\mathbf{A}}^j}{\operatorname{argmin}} \sum_{j \in \{0,1\}} - \int_{\mathbf{Z}'^t} \alpha_j \log\left(d_{\mathbf{A}}^j\left(r(\mathbf{Z}'^t)\right)\right) P_j^t d\mathbf{Z}'^t. \tag{14}$$

We can also take pointwise optimization for any $\mathbf{Z}'^t$ in Eq. (14) Then by combining Equation (14) with the constraint in Equation (12) and using Lagrange multipliers, we have:

$$d_{\mathbf{A}}^{j\,*} = \underset{d_{\mathbf{A}}^j}{\operatorname{argmin}} \sum_{j \in \{0,1\}} -\alpha_j \log\left(d_{\mathbf{A}}^j\left(r(\mathbf{Z}'^t)\right)\right) P_j^t + \lambda\left(\sum_{j \in \{0,1\}} d_{\mathbf{A}}^j\left(r(\mathbf{Z}^t)\right) - 1\right). \tag{15}$$

Let $J = \sum_{j \in \{0,1\}} -\alpha_j \log(d_{\mathbf{A}}^j(r(\mathbf{Z}'^t)))P_j^t + \lambda(\sum_{j \in \{0,1\}} d_{\mathbf{A}}^j(r(\mathbf{Z}^t)) - 1)$ be the objective in Eq. (15) The optimal values can be obtained by taking partial gradients $\frac{\partial J}{\partial d_{\mathbf{A}}^j} = 0$ and $\frac{\partial J}{\lambda} = 0$, respectively. By computing them jointly we can obtain $d_{\mathbf{A}}^{j\,*} = \frac{\alpha_j P_j^t}{\sum_{j' \in \{0,1\}} \alpha_{j'} P_{j'}^t}$. □

The second step is to prove Theorem. 1 that the optimal f and ϕ can obtain balanced representations with respect to treatments.

Proof. With Proposition 1, we can fix the optimal $d_{\mathbf{A}}^{j\,*}$ and find the condition where Eq. (7) achieves optimum. Putting $d_{\mathbf{A}}^{j\,*}$ into the objective in Eq. (7) and applying similar simplifications as in Eq. (14) and Eq. (15), we have:

$$f^*, \phi^* = \underset{f,\phi}{\operatorname{argmax}} \sum_{j \in \{0,1\}} -\mathbb{E}_{\mathbf{Z}^t \sim P_j^t} \log\left(\frac{\alpha_j P_j^t}{\sum_{j' \in \{0,1\}} \alpha_{j'} P_{j'}^t}\right) \tag{16}$$

$$= \underset{f,\phi}{\operatorname{argmin}} \sum_{j \in \{0,1\}} \mathbb{E}_{\mathbf{Z}^t \sim P_j^t} \log\left(\frac{P_j^t}{\sum_{j' \in \{0,1\}} \alpha_{j'} P_{j'}^t}\right) + \log(\alpha_j) \tag{17}$$

$$= \underset{f,\phi}{\operatorname{argmin}} \sum_{j \in \{0,1\}} \mathrm{KL}\left(P_j^t \middle\| \sum_{j' \in \{0,1\}} \alpha_{j'} P_{j'}^t\right) + \log(\alpha_j), \tag{18}$$

(19)

where $\mathrm{KL}(\cdot||\cdot)$ is the KL divergence. Note that $\sum_{j \in \{0,1\}} \log(\alpha_j)$ is constant in observation data. $\mathrm{KL}(\cdot||\cdot) \geq 0$ and it reaches 0 when the two operands are equal. Therefore, to have f^*, ϕ^*, for $j \in \{0,1\}$, we have $P_0^t = P_1^t = \sum_{j' \in \{0,1\}} \alpha_{j'} P_{j'}^t$. Therefore, the optimal f^*, ϕ^* are those who achieve $P_0^t = P_1^t$. We can apply this to all the N units and all the T observed timestamps to obtain the global optimum of Eq. (7), which concludes the proof of Theorem 1. □

A.1.2 Proof of Theorem 2.

Theorem 2. *Let f, ϕ, d_{G} be the initial state encoder, the ODE function of treatment-induced GraphODE, and the interference predictor. The necessary and sufficient condition for min-max game in Eq. (8) to be optimal is $P(\mathbf{Z}^t, \mathbf{A}^t | \mathrm{G}^t = g')$ is identical for any g'.*

The proof of Theorem 2 follows [52]. Similar to Theorem 1's proof, it first finds the optimum of d_{G}, and then proves that the optimal f and ϕ can balance the representations with respect to interference. We first restate the Lemma 4.1 in [52] in Proposition. 2.

Proposition 2. *(Lemma 4.1 in [52]) Let* $\mathbf{C}^t = [\mathbf{Z}^t, \mathbf{A}^t]$ *be the concatenation of* $\mathbf{Z}^t$ *and* $\mathbf{A}^t$*. When fixing initial state encoder* f *and ODE function* ϕ*, the optimal* d_{G} *at time* t *is:*

$$d^*_{\mathrm{G}} = \mathbb{E}_{\mathbf{G}^t \sim p(\mathbf{G}^t|\mathbf{C}^t)}(\mathbf{G}^t). \tag{20}$$

Proof. Eq. (20) is adapted from Eq. (8). We disregard the expectation with respect to i and t since Eq. (20) is applicable to any i and t. If fixing f and ϕ, the optimal d^*_{G} is given by:

$$d^*_{\mathrm{G}} = \underset{d_{\mathrm{G}}}{\operatorname{argmin}}\, \mathbb{E}_{(\mathbf{Z}^t,\mathbf{A}^t,\mathbf{G}^t)\sim p(\mathbf{Z}^t,\mathbf{A}^t,\mathbf{G}^t)} \left(d_{\mathrm{G}}\left(r([\mathbf{Z}^t, \mathbf{A}^t])\right) - \mathbf{G}^t\right)^2 \tag{21}$$

$$= \underset{d_{\mathrm{G}}}{\operatorname{argmin}}\, \mathbb{E}_{(\mathbf{C}^t,\mathbf{G}^t)\sim p(\mathbf{C}^t,\mathbf{G}^t)} \left(d_{\mathrm{G}}\left(r(\mathbf{C}^t)\right) - \mathbf{G}^t\right)^2 \tag{22}$$

$$= \underset{d_{\mathrm{G}}}{\operatorname{argmin}}\, \mathbb{E}_{\mathbf{C}^t\sim p(\mathbf{C}^t)}\mathbb{E}_{\mathbf{G}^t\sim p(\mathbf{G}^t|\mathbf{C}^t)} \left(d_{\mathrm{G}}\left(r(\mathbf{C}^t)\right) - \mathbf{G}^t\right)^2. \tag{23}$$

As the quadratic expansion in [52], the optimal interference predictor is $d^*_{\mathrm{G}} = \mathbb{E}_{\mathbf{G}^t\sim p(\mathbf{G}^t|\mathbf{C}^t)}(\mathbf{G}^t)$. □

Here we introduce and reformulate Theorem 4.1 in [52] as Lemma. 1.

Lemma 1. *(Theorem 4.1 in [52]) Given* $\mathbb{E}_x\mathbb{V}(y \mid x)$ *where* $\mathbb{V}$ *denotes variance, its global optimum can be achieved if and only if for any* x, $\mathbb{E}(y \mid x) = \mathbb{E}(y)$.

With Proposition. 2 and Lemma. 1, we can prove Theorem. 2.

Proof. Fixing the optimal d^*_{G} in Proposition. 2, the optimal f and ϕ for objective Eq. (23) is:

$$f^*, \phi^* = \underset{f,\phi}{\operatorname{argmax}}\, \mathbb{E}_{\mathbf{C}^t\sim p(\mathbf{C}^t)}\mathbb{E}_{\mathbf{G}^t\sim p(\mathbf{G}^t|\mathbf{C}^t)} \left(\mathbb{E}_{\mathbf{G}^t\sim p(\mathbf{G}^t|\mathbf{C}^t)}(\mathbf{G}^t) - \mathbf{G}^t\right)^2 \tag{24}$$

$$= \underset{f,\phi}{\operatorname{argmax}}\, \mathbb{E}_{\mathbf{C}^t\sim p(\mathbf{C}^t)}\mathbb{V}\left(\mathbf{G}^t \mid \mathbf{C}^t\right). \tag{25}$$

Eq. (25) has the same form as the equation in Lemma. 1. Then substituting $x = \mathbf{C}^t$ and $y = \mathbf{G}^t$ in Lemma. 1, we have $\mathbb{E}(\mathbf{G}^t \mid \mathbf{C}^t) = \mathbb{E}(\mathbf{G}^t)$. In other words, $\mathbf{G}^t \perp \mathbf{C}^t$. Therefore, we can also use the inverse form that $\mathbb{E}(\mathbf{C}^t) = \mathbb{E}(\mathbf{C}^t \mid \mathbf{G}^t) = \mathbb{E}([\mathbf{Z}^t, \mathbf{A}^t] \mid \mathbf{G}^t)$ for any $\mathbf{G}^t$, which concludes the proof of Theorem. 2. □

A.2 Pseudo-Code of CF-GODE Training

The pseudo-code for training CF-GODE is shown in Algorithm. 1. Specifically, we use an alternative training trick to trade-off the outcome prediction and adversarial balancing.

Algorithm 1 The optimization process of CF-GODE

```
Input: Multi-agaent dynamical system $\mathcal{G}$; the observational data $\left((\mathbf{X}^t, \mathbf{A}^t, \mathbf{Y}^t) \cup \mathbf{V}\right)$; observed timestamps $(t_0, t_1 \cdots t_T)$.
Output: Trained initial state encoder $f(\cdot)$; treatment-induced GraphODE function $\phi(\cdot)$; outcome decoder $d_{\mathrm{Y}}(\cdot)$; treatment predictor $d_{\mathrm{A}}(\cdot)$; interference predictor $d_{\mathrm{G}}(\cdot)$.
Training:
Initialize $f(\cdot)$, $\phi(\cdot)$, $d_{\mathrm{Y}}(\cdot)$, $d_{\mathrm{A}}(\cdot)$, $d_{\mathrm{G}}(\cdot)$;
for w = 1, 2, ..., W do                                  ▷ Train W iterations
    if w%(K+1) = 0 then                                  ▷ Train with $L^{\langle Y \rangle}$ for 1 steps
        Compute $L^{\langle Y \rangle}$;
        One step optimization for $f(\cdot)$, $\phi(\cdot)$, and $d_{\mathrm{Y}}(\cdot)$:
        $\theta_f^{(w+1)} = \theta_f^{(w)} - \eta\nabla_{\theta_f} L^{\langle Y \rangle}$;          ▷ $\eta$ is learning rate
        $\theta_\phi^{(w+1)} = \theta_\phi^{(w)} - \eta\nabla_{\theta_\phi} L\langle Y \rangle$;
        $\theta_{d_{\mathrm{Y}}}^{(w+1)} = \theta_{d_{\mathrm{Y}}}^{(w)} - \eta\nabla_{\theta_{d_{\mathrm{Y}}}} L^{\langle Y \rangle}$;
    else                                                 ▷ Train with $L$ for $K$ steps
        Compute $L$;
        One step optimization for $f(\cdot)$, $\phi(\cdot)$, $d_{\mathrm{Y}}(\cdot)$, $d_{\mathrm{A}}(\cdot)$, $d_{\mathrm{G}}(\cdot)$:
        $\theta_{d_{\mathrm{A}}}^{(w+1)} = \theta_{d_{\mathrm{A}}}^{(w)} - \eta\nabla_{\theta_{d_{\mathrm{A}}}} L$;
        $\theta_{d_{\mathrm{G}}}^{(w+1)} = \theta_{d_{\mathrm{G}}}^{(w)} - \eta\nabla_{\theta_{d_{\mathrm{G}}}} L$;
        $\theta_{d_{\mathrm{Y}}}^{(w+1)} = \theta_{d_{\mathrm{Y}}}^{(w)} - \eta\nabla_{\theta_{d_{\mathrm{Y}}}} L$
        $\theta_f^{(w+1)} = \theta_f^{(w)} - \eta\nabla_{\theta_f} L^{\langle Y \rangle} + \eta\nabla_{\theta_f} L^{\langle A \rangle} + \eta\nabla_{\theta_f} L^{\langle G \rangle}$;
                                                         ▷ Gradient reversal
        $\theta_\phi^{(w+1)} = \theta_\phi^{(w)} - \eta\nabla_{\theta_\phi} L^{\langle Y \rangle} + \eta\nabla_{\theta_\phi} L^{\langle A \rangle} + \eta\nabla_{\theta_\phi} L^{\langle G \rangle}$;
                                                         ▷ Gradient reversal
Return $f(\cdot)$, $\phi(\cdot)$, $d_{\mathrm{Y}}(\cdot)$, $d_{\mathrm{A}}(\cdot)$, $d_{\mathrm{G}}(\cdot)$.
```

A.3 Experimental Settings

Datasets. In observational data, we only have outcomes under one treatment trajectory but not the ground-truths of counterfactual outcomes. Therefore, we follow [16, 29, 50] to use semi-synthetic data to evaluate CF-GODE. That is, the graph structure and node features are real, but treatments and potential outcomes are simulated. We use the social networks Flickr and BlogCatalog as in [6, 16, 28] as the graph $\mathcal{G}$. We follow these works to first encode the node features into low-dimensional embeddings (10-dimensional in this paper) via LDA [4]. We then follow [21] to use Metis [23] to split the graph into training/validation/testing sets. To simulate treatment and potential outcomes over time, [3, 14, 46] use a longitudinal simulation environment, which, however, assumes the units are mutually independent. We extend it into our multi-agent dynamical systems setting by considering the neighbor confounders and interference. During the simulation, we are motivated by the vaccine's use case. Specifically, treatment $\mathbf{A}_i^t$ denotes getting a vaccine or not at time t of unit i. The trajectory of $\mathbf{A}_i^t$ denotes the vaccine records over time, e.g., a unit may have a booster dose after the initial vaccine. In this case, the time-dependent covariates $\mathbf{X}^t$ could be the health condition, static covariates $\mathbf{V}$ could be race or educational background (assuming it does not change during the study) and potential outcome $\mathbf{Y}^t$ could be immunity to the virus. As discussed in Sec. 3.1, the potential outcome $\mathbf{Y}^t$ is essentially a part of $\mathbf{X}^t$. This is a common setting in longitudinal causal inference studies [3, 33, 46].

During the simulation, we follow this protocol: the health condition $\mathbf{X}_i^t$ has a value range $[0.1, 10]$, in which a higher value means a better health condition. Meanwhile, a higher health condition means a lower probability to receive a vaccine (treatment).

Treatment simulation. The treatment $\mathbf{A}_i^t$ is affected by a unit's own time-dependent covariates $\mathbf{X}_i^t$, static covariates $\mathbf{V}_i$ and those of their neighbors. Let $\mathbf{E}_i = w_a \mathbf{V}_i$ denote the effects of static confounders on treatments, where w_a is a generated parameter representing this mechanism. The treatment is then simulated by Bernoulli generator with probability the $p_i^t(a)$ of unit i at time t:

$$p_i^t(a) = \sigma\Bigg(\underbrace{\gamma_a(\delta_a - \bar{\mathbf{X}}_i^t)}_{\text{time-dependent covariates}} + \underbrace{\gamma_n\Bigg(\delta_n - (\frac{1}{|N_i|}\sum_{j\in\mathcal{N}_i}\bar{\mathbf{X}}_j^t)\Bigg)}_{\text{Neighbor time-dependent covariates}} + \underbrace{\gamma_f \mathbf{E}i}_{\text{Static covariates}} + \underbrace{\gamma_g(\frac{1}{|N_i|}\sum_{j\in\mathcal{N}_i}\mathbf{E}_j)}_{\text{Neighbor static covariates}} \Bigg), \tag{26}$$

where $\sigma(\cdot)$ is sigmoid function. $\gamma_a, \gamma_n, \gamma_f$ and γ_g are degrees of time-dependent confounders, neighbor time-dependent confounders, static confounders and neighbor static confounders, respectively. The default values are $[\gamma_a, \gamma_n, \gamma_f, \gamma_g] = [10, 3.3, 10, 3.3]$ ($\frac{\gamma_a}{\gamma_n} = \frac{\gamma_f}{\gamma_g} = 3$), to mimic that a unit's own confounding factors should affect it more than neighbors. Note $\bar{\mathbf{X}}_i^t$ is the average time-dependent covariates until t. This reflects that past time-dependent covariates also affect the treatment. We set $\delta_a = \delta_n = 5$ as adjustments.

Potential outcome simulation. We follow [3, 14, 46] to use a Pharmacokinetic-Pharmacodynamic (PK-PD) model [15] to simulate the continuous trajectory. PK-PD model is a popular bio-mathematical model and a natural fit for our vaccine use case. As mentioned above, $\mathbf{Y}^t$ is essentially a part of $\mathbf{X}^t$. Therefore we directly simulate the trajectory of $\mathbf{X}_i^t$:

$$\frac{d\mathbf{X}_i^t}{dt} = \mathbf{X}_i^t\Bigg(\underbrace{\rho_u \log\left(\frac{K}{\mathbf{X}_i^t}\right)}_{\text{Time-dependt covariates}} + \underbrace{\rho_n \log\left(\frac{K}{\mathbf{X}_i^t}\right)}_{\text{Neighbor time-dependt covariates}} + \underbrace{\rho_f \mathbf{O}_i}_{\text{Static covariates}} + \underbrace{\rho_g \sum_{j\in\mathcal{N}_i}\mathbf{O}_j)}_{\text{Neighbor static covariates}} + \underbrace{\beta_a \mathbf{D}_i^t}_{\text{Treatment}} + \underbrace{\frac{1}{|N_i|}\sum_{j\in\mathcal{N}_i}\beta_n \mathbf{D}_j^t}_{\text{Interference}} + \underbrace{e_i^t}_{\text{Noise}} \Bigg), \tag{27}$$

where K controls the effects of time-dependent covariates on future potential outcomes. $\mathbf{O}_i = w_x \mathbf{V}_i$ denote the effects of static confounders on potential outcomes, where w_x represents this mechanism. ρ_u, ρ_n, ρ_f and ρ_g are degrees of time-dependent covariates, neighbor time-dependent covariates, static covariates and neighbor static covariates, respectively. Their default values are $[\rho_u, \rho_n, \rho_f, \rho_g] = [-0.001, -00033, 0.001, 0.00033]$ ($\frac{\rho_u}{\rho_n} = \frac{\rho_f}{\rho_g} = 3$). β_a and β_n control the strengths of treatment and interference. We set them as $[\beta_a, \beta_n] = [0.03, 0.01]$. The values also reflect that a unit's own covariates and treatment should have stronger effects on its future potential outcomes than neighbors. In reality, the effects of vaccines on providing protection decrease over time. To mimic this phenomenon, we use the following decay function to model the effects of treatments over time as in [3, 46].

$$\mathbf{D}_i^t = \tilde{\mathbf{D}}_i^t + \mathbf{D}_i^{(t-1)}/2, \tag{28}$$

where $\mathbf{D}_i^t$ denote the protection effect at time t, and $\tilde{\mathbf{D}}_i^t$ means a full protection of vaccines given at t. In other words, the unit receives a vaccine at time t, i.e., $\mathbf{A}_i^t = 1$. We set $\tilde{\mathbf{D}}_i^t = 1$.

A.4 Notation Table

The notations and their corresponding description are in Table. 4.

Table 4: Notations.

Notation	Description
$\mathcal{G}$	Graph that represents multi-agent dynamical system
$\mathcal{V}$	Node set in $\mathcal{G}$
$\mathcal{E}$	Edge set in $\mathcal{G}$
$\mathbf{V}$	Static unit features
$\mathbf{X}^t$	Time-dependent covariates at time t
$\mathbf{A}^t$	Treatment at time t
$\mathbf{Y}^t$	Observed outcomes at time t
$\mathbf{Y}^t(A^t = a)$	Potential outcomes under treatment a
$\bar{\mathbf{X}}^t$	Time-dependent covariates collections up to t
$\bar{\mathbf{A}}^t$	Treatment collections up to t
$\bar{\mathbf{Y}}^t$	Observed outcomes collections up to t
$\mathcal{H}^t$	Past observations
$\mathbf{A}_{\mathcal{N}_i}^t$	Treatments of node i's first-order neighbors
$\mathbf{A}_{\mathcal{N}_{-i}}^t$	Treatments of nodes that are beyond i's first-order neighbors
$\mathbf{G}_i^t$	Interference summary variable
$\mathbf{Z}_i^t$	Continuous latent trajectory for node i
$\mathbf{C}_i^t$	Concatenation of $\mathbf{Z}_i^t$ and $\mathbf{A}_i^t$
$g(\cdot)$	Interference summary function
$\phi(\cdot)$	ODE function
$f(\cdot)$	Initial state encoder function
$r(\cdot)$	Gradient reversal layer
$d_{\mathbf{Y}}(\cdot)$	Outcome prediction layer
$d_{\mathbf{A}}(\cdot)$	Treatment prediction layer
$d_{\mathbf{G}}(\cdot)$	Interference prediction layer
N	Number of nodes in $\mathcal{V}$
T	Number of observed timestamps
$L^{\langle Y\rangle}$	Loss function of outcome prediction
$L^{\langle A\rangle}$	Loss function of treatment prediction
$L^{\langle G\rangle}$	Loss function of interference prediction
$\alpha_{\mathbf{A}}$	Weight of treatment balancing
$\alpha_{\mathbf{G}}$	Weight of interference balancing