Causally Consistent Normalizing Flow

Qingyang Zhou[1], **Kangjie Lu**[2], **Meng Xu**[1]

[1]University of Waterloo, Ontario, Canada
[2]University of Minnesota, Minnesota, America
qingyang.zhou@uwaterloo.ca, kjlu@umn.edu, meng.xu.cs@uwaterloo.ca

Abstract

Causal inconsistency arises when the underlying causal graphs captured by generative models like Normalizing Flows are inconsistent with those specified in causal models like Struct Causal Models. This inconsistency can cause unwanted issues including the unfairness problem. Prior works to achieve causal consistency inevitably compromise the expressiveness of their models by disallowing hidden layers. In this work, we introduce a new approach: Causally Consistent Normalizing Flow (CCNF). To the best of our knowledge, CCNF is the first causally consistent generative model that can approximate any distribution with multiple layers. CCNF relies on two novel constructs: a sequential representation of SCMs and partial causal transformations. These constructs allow CCNF to inherently maintain causal consistency without sacrificing expressiveness. CCNF can handle all forms of causal inference tasks, including interventions and counterfactuals. Through experiments, we show that CCNF outperforms current approaches in causal inference. We also empirically validate the practical utility of CCNF by applying it to real-world datasets and show how CCNF addresses challenges like unfairness effectively.

Code — https://github.com/UWCSZhou/CCNF
Extended version — https://arxiv.org/abs/2412.12401

1 Introduction

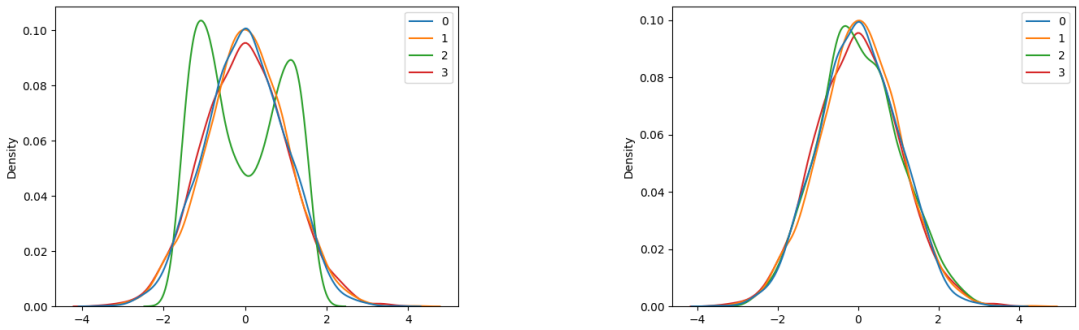
Causal generative modeling is generative models (GMs) that utilize given causal models like structure causal models (SCMs) for data generation (Komanduri et al. 2024). It has been widely researched on different types of GMs like VAE (Yang et al. 2021), GAN (Kocaoglu et al. 2017), Normalizing Flow (NF) (Javaloy, Martin, and Valera 2023) and Diffusion Model (Sanchez and Tsaftaris 2022).

However, most approaches have a problem that they can only approximate the causality relations instead of *enforcing* the consistency between the causal graph induced by GMs and the causal graph in given SCMs. The problem is called casual inconsistency problem in prior works (Javaloy, Martin, and Valera 2023) and will be discussed in detail in Section 3. This could lead to critical societal issues (e.g. the one in Section 2), which have yet to be adequately addressed.

Fortunately, recent works have proposed GMs that are causally consistent by design. Causal NF (Javaloy, Martin, and Valera 2023) and VACA (Sánchez-Martin, Rateike, and Valera 2022) ensure causal consistency by limiting their models to minimal structural complexity. More specifically, Causal NF restricts the model depth to zero, i.e., eliminating middle layers in NF; while VACA applies a similar restriction to its encoder structure. In other words, causal consistency is guaranteed at the expense of the utility of these models—the ability to approximate any arbitrarily complex distributions of observations. For instance, the training objective of Causal NF is to minimize the discrepancy between the distributions of latent variables and the pre-selected distributions by users (e.g. Gaussian). However, as shown in Figure 1a, a Causal NF trained on a nonlinear Simpson dataset is not able to accomplish the objective. The distribution of the third latent variable, highlighted in green in Figure 1a, deviates significantly from the Gaussian distribution.

In this paper, we introduce **C**ausally **C**onsistent **N**ormalizing **F**low, abbreviated as CCNF, that is a causally consistent GM by design without sacrificing utility, i.e., approximating arbitrarily complex distributions based on universal approximation theorems (details in Theorem 5.2). A key innovation of CCNF is to translate an SCM into a sequence (details in Secion 4). The sequential representation of an SCM eliminates the constraints of maximum layer depths without compromising causal consistency, enabling a more flexible model architecture in CCNF (details in Section 5). Subsequently, CCNF employs normalizing flows with partial causal transformations to effectively capture the causality in the data, which is further elaborated in Section 4 and Section 5.

We demonstrate that CCNF is inherently causally consistent and capable of performing causal inference tasks such as interventions and counterfactuals. To the best of our knowledge, CCNF is the first causally consistent GM that can approximate any distributions of observation variables across multiple layers. In comparison, CCNF outperforms existing models like Causal NF in similar tasks, as shown in 1b. Additionally, CCNF proves effective in real-world applications, addressing significant issues such as unfairness. Applying CCNF to the German credit dataset (Hofmann 1994), we observe notable improvements: a reduction in individual unfairness from 9.00% to 0.00%, and an increase in overall



(a) causal NF (b) CCNF

Figure 1: Prior distributions of causal NF and CCNF

accuracy from 73.00% to 75.80%.

Summary. This paper makes the following contributions:

- We propose a new sequential representation for SCMs, and formally prove its ability to maintain the causal consistency.
- Utilizing this sequential representation alongside partial causal transformations, we develop CCNF, a GM that guarantees causal consistency and excels at complex causal inference tasks.
- We empirically validate CCNF, demonstrating it outperforms state-of-the-art casually consistent GMs on causal inference benchmarks. Furthermore, our real-world case study showcase the potential of CCNF in addressing critical issues like unfairness. CCNF is open-sourced in the code link.

2 A Motivating Example

To articulate the causal inconsistency problem between GMs and SCMs, we present a motivating example modeling a simplified admission system.

A simplified admission system. The causal graph of a simplified admission system $\mathcal{M}$ is shown in Figure 2. It consists of four attributes: `gender`, `age`, `score`, and admission `decision`. In terms of casualty, `gender` and `age` determine the distribution of `score`, and ideally, `score` solely determines the distribution admission `decision`, ensuring that `gender` does not (and should not) directly affect admission.

To illustrate, we assume the observations $\mathbf{O}$ of this admission system $\mathcal{M}$ can be generated by the equations in Table 1 under the SCM column. Here, the value of independent variables u_i where $i \in \{g, a, s, d\}$ are randomly sampled from predefined distributions. For instance, u_g is sampled from a distribution where gender is distributed equally. With distinct samples for u_i where $i \in \{g, a, s, d\}$, this system can generate unique observations representing $\mathbf{O}$.

GMs based on the admission system. In this scenario, $\mathcal{GM}$s learned from the observations $\mathbf{O}$ offer extensive capabilities (Harshvardhan et al. 2020). However, a $\mathcal{GM}$ may establish incorrect causal links, such as between `gender` and admission `decision`, as indicated by the red dashed line in Figure 2. This incorrect causality suggests that in $\mathcal{GM}$, different values of `gender` could lead to varied distributions of admission `decision`, even if the `score` is identical.

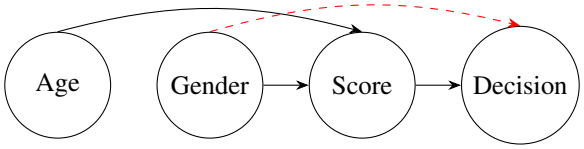


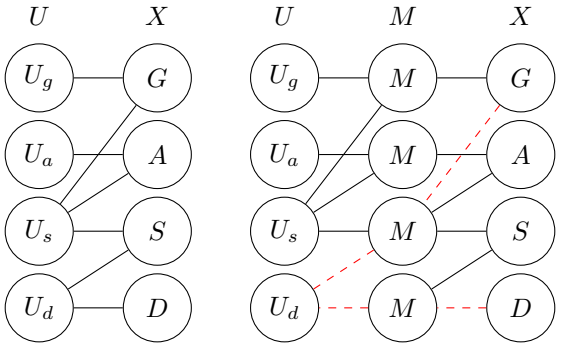
Figure 2: The causal graph of an SCM describing an admission system with direct causalities that are intended (black solid line) and forbidden (red dashed line)

Variable	SCM	GM
`gender`	u_g	u_g
`age`	u_a	u_a
`score`	u_s+`gender`+`age`	u_s-`gender`+`age`
`decision`	$f(u_d$+`score`$)$	$f(u_d$+`score`+2*`gender`$)$

Table 1: Comparison of generative equations between the actual SCM and GM. The function f in the table stands for the $sign$ function.

Consider the scenario where the underlying causal relationships of a $\mathcal{GM}$ are formulated as in the GM column of the Table 1. We can verify that the data distributions generated by the $\mathcal{GM}$ is indistinguishable from those of $\mathcal{M}$, despite in distinct functional forms. However, $\mathcal{GM}$ and $\mathcal{M}$ are not causally consistent, which could result in significant consequences.

Unfairness due to causal inconsistency. In this example, causal inconsistency could cause an unfairness problem. Consider a scenario where users generate data instances with the same `score`. In the origin system $\mathcal{M}$, as `gender` has no direct impact on admission `decision`, users will observe that changing the gender attribute does not alter the distribution of admission `decision`. However, in the causally inconsistent model $\mathcal{GM}$, users will surprisingly observe that different values of `gender` could lead to different distribu-



(a) Consistent (b) Inconsistent

Figure 3: Causally consistent models and inconsistent models of prior works. *G* = *Gender*, *A* = *age*, *S* = *Score*, *D* = *Decisions*, *M* stands for nodes in the middle layer.

tions of `decision`, even with the same score. Such gender bias could lead to intense social debates and the organization deploying the $\mathcal{GM}$ might face legal challenges (Thompso 2023).

Prior works and their drawbacks. Despite being applied to various architectures of GMs, prior works on guaranteed causally consistent GMs share a common intuition: while each attribute in observation is influenced only by its parent attributes within the causal graph, the causality relation must be captured in the GM model **all at once**, which means that their layer depth must be zero. Figure 3a shows an example of this approach. In this scenario, if we maintain constant values for `age` and `score` and only mutate the value of `gender`, The distribution of admission `decision` should remain unaffected.

This intuition indeed ensures causal consistency, but it has some shortcomings; notably, it does not allow for incorporation of any middle layers. For example, mutating the value of `gender` in Figure 3b results in a change of admission `decision` due to the red dashed connections. This comparison between Figure 3b and Figure 3a illustrates how causal consistency is compromised even with the introduction of a single layer in the middle, while on the other hand, forbidding middle layers significantly impairs the model's capacity for learning. However, as we will demonstrate later, CCNF can maintain causal consistency even with multiple middle layers. Therefore CCNF offers great learning ability compared to previous models, enhancing practical applicability.

3 Preliminaries

In this section, we define basic concepts and related lemmas to set the stage for CCNF. All definitions and lemmas introduced in this section are consistent with prior works (Khemakhem et al. 2021; Papamakarios et al. 2021; Thost and Chen 2021).

Structured Casual Model (SCM)

Definition. A structural causal model (SCM) is a tuple $\mathcal{M} = (\widetilde{\mathbf{f}}, P_{\mathrm{u}})$ commonly used to represent causality. It describes the process that a set of d endogenous (observed) random variables $\mathbf{X} = \{X_1, \cdots, X_d\}$ is generated from a corresponding set of exogenous (latent) random variables $\mathbf{U} = \{U_1, \cdots, U_d\}$ associated with a set of predefined distributions P_{u} and a set of transfer functions $\widetilde{\mathbf{f}}$. Typically, $\mathbf{X}$ and $\mathbf{U}$ have the same length, denoting as d. The generation of $\mathbf{X}$ is governed by the equation:

$$X_i = \widetilde{f}_i(\mathbf{X}_{\mathbf{pa}_i}, U_i), \quad i \in \{1, \ldots, d\} \tag{1}$$

where $\mathbf{X}_{\mathbf{pa}_i}$ denotes a set of endogenous variables that *directly* influences X_i, i.e. the parents of X_i. Particularly, $\mathbf{pa}_i$ represents the labels of those parent variables of X_i. Generally, we assume exogenous variables $\mathbf{U}$ are mutually independent.

Graphical representation of an SCM: causal graph. A causal graph $\mathcal{G} = (\mathbf{V}, \mathbf{E})$ with $|\mathbf{V}| = d$ nodes representing an SCM is a powerful tool to describe causality. Each node in $\mathbf{V}$ corresponds to an endogenous random variable X_i, and each edge in $\mathbf{E}$ represents a causal link from a variable in $\mathbf{X}_{\mathbf{pa}_i}$ to X_i. As a common assumption (Pearl 2009), the causal graph $\mathcal{G}$ is structured as a directed acyclic graph (DAG). For convenience, each node $V \in \mathbf{V}$ is assigned with a label $i \in \mathbf{L} = \{1, \cdots, d\}$, and we denote it as V_i. For a subset of labels $\mathbf{A} \subseteq \mathbf{L}$, we define $\mathbf{V}_{\mathbf{A}} = \{V_i \mid i \in \mathbf{A}\}$. See Figure 4a for an illustration.

Causal inference and causal hierarchy. *Causal inference* denotes the data generation process by SCM. According to the *causal hierarchy* (Pearl 2009), the generative process is classified into three distinct tiers: *observations*, *interventions*, and *counterfactuals*. The *observations* process involves generating $\mathbf{X}$ unconditionally. This is straightforward: we generate $\mathbf{U}$ sampled from the predefined distributions P_{u} and then compute $\mathbf{X}$ from $\mathbf{U}$ by the formula in Equation 1. The *interventions* process involves generating $\mathbf{X}$ while setting the X_i to a specific value of a, often represented as $Do(X_i = a)$. This requires modifying the SCM such that every X_i in Equation 1 is replaced with a to create a new SCM: $\mathcal{M}_{Do(X_i=a)}$. $\mathbf{X}$ is generated by performing observations on the new SCM. The *counterfactuals* process considers a specific data instance $\mathbf{X}$ where $X_j = b$, and aims to generate data instances supposing that $X_j = b' \neq b$. This process first deduces $\mathbf{U}$ from $\mathbf{X}$ via Equation 1, then performs $Do(X_j = b')$ to create a new SCM $\mathcal{M}_{Do(X_j=b')}$, Subsequently, the data is generated by inputting $\mathbf{U}$ into $\mathcal{M}_{Do(X_i=a)}$.

Causal Normalizing Flows

Normalizing flows. *Normalizing flows* constitute a set of generative models that express the probability of observed variables $\mathbf{X}$ from $\mathbf{U}$ by change-of-variables rules. Particularly, given $\mathbf{X} = \{X_1, \cdots, X_d\}$ and $\mathbf{U} = \{U_1, \cdots, U_d\}$, the probability of $\mathbf{X}$ is expressed as follows:

$$\mathbf{X} = T_\theta(\mathbf{U}), \quad \textit{where } \mathbf{U} \sim P_{\mathbf{U}} \tag{2}$$

$$P_{\mathbf{X}}(\mathbf{X}) = P_{\mathbf{U}}(T_\theta^{-1}(\mathbf{X}))|\det J_{T_\theta^{-1}}(\mathbf{X})| \tag{3}$$

Here, T_θ represents a transformation that maps endogenous variables $\mathbf{U}$ to exogenous variables $\mathbf{X}$. T_θ could be any transformation as long as it is a partial derivative and invertible, often realized by a neural network with parameter θ. It is common to chain different transformations $T_{\theta_1} \cdots T_{\theta_k}$ to form a larger transformation $T_\theta = T_{\theta_k} \circ \cdots T_{\theta_2} \circ T_{\theta_1}$. $P_{\mathbf{X}}$ denotes the probability of $\mathbf{X}$ while $P_{\mathbf{U}}$ denotes the probability of $\mathbf{U}$, where $P_{\mathbf{X}}$ is the target and $P_{\mathbf{U}}$ usually follows a simple distribution. The $\det J$ means the Jacobian determinant of a given function, which is T_θ^{-1} in this formula.

Multi-layer universal approximator. An NF serves as a *multi-layer universal approximator*, implying that any $P_{\mathbf{X}}$ can be approximated by chaining a finite number of transformations. Comparatively, the single-layer universal approximator can achieve the same objective with only one transformation. The assumption that a NF is single-layer universal is stronger than the assumption that it is multi-layer.

Although an NF the theoretical capability is single-layer universal (Papamakarios et al. 2021), No concrete NF succeeded in proving this. Instead, a recent study on the university of coupling-based NF proves that affine coupling flows like MAF (Papamakarios, Pavlakou, and Murray 2018) are multi-layer universal (Draxler et al. 2024).

Autoregressive normalizing flows. *Autoregressive normalizing flows* are a type of NFs whose transformations are defined as below: given two random variables $\mathbf{U} = \{U_1, \cdots, U_d\}$ and $\mathbf{X} = \{X_1, \cdots, X_d\}$, the special transformation of Equation 2 is:

$$X_i = T_\theta(U_i \mid \mathbf{X}_{<i}), \quad i \in \{1, \ldots, d\} \tag{4}$$

Here, $\mathbf{X}_{<i} = \{X_1, \cdots, X_{i-1}\}$. This formula indicates that the parameter θ in $T_\theta(U_i \mid \mathbf{X}_{<i})$ is determined by $\mathbf{X}_{<i}$, and the output value X_i is directly determined only by U_i. This transformation yields a simple Jacobin determinant since its Jacobin matrix is lower triangular.

Autoregressive flows and causality. Although many works (Pawlowski, Coelho de Castro, and Glocker 2020; Ribeiro et al. 2023) have utilized NFs for causal inference tasks like counterfactual inference, autoregressive flows and casualty were still considered as two unrelated fields. However, Ilyes noticed that it is possible to leverage autoregressive flows for causal tasks due to their intrinsic similarity (Khemakhem et al. 2021). Particularly, to capture the causal relation between X_i and U_i accurately, autoregressive flows possess a transformation as follows:

$$X_i = T_\theta(U_i \mid \mathbf{X}_{\mathbf{pa}_i}), \quad i \in \{1, \ldots, d\} \tag{5}$$

In contrast to Equation 4, the parameter θ in Equation 5 is determined by $\mathbf{X}_{\mathbf{pa}_i}$ rather than $\mathbf{X}_{<i}$.

Causal Consistency

For any given SCM $\mathcal{M}$ and the its casual graph $\mathcal{G}_\mathcal{M}$, we call a $\mathcal{GM}$ is causally consistent with $\mathcal{M}$ if the causal graph $\mathcal{G}_{\mathcal{GM}}$ induced by the $\mathcal{GM}$ is the same as $\mathcal{G}_\mathcal{M}$. According to previous work, the special $\mathcal{GM}$s exist and can produce consistent result in all three tiers of causal hierarchy (Xia, Pan, and Bareinboim 2022).

Note that causal consistency only require the $\mathcal{M}$ and $\mathcal{G}_\mathcal{M}$ to share the same causal graph. In previous work (Xi and Bloem-Reddy 2023), they have proved that such $\mathcal{GM}$ and $\mathcal{M}$ are identifiable, which means the data-generating process between $\mathcal{GM}$ and $\mathcal{M}$ only differs by an invertible component-wise transformation of the variables in $\mathbf{U}$, therefore $\mathcal{GM}$ can perform causal inference tasks just like $\mathcal{M}$.

Topological Batching

Topological batching results in an ordered sequence $\mathbf{B} = (\mathbf{B}_1, \cdots, \mathbf{B}_n)$ which partitions the label set $\{1, \ldots, d\}$ of a DAG $\mathcal{G} = (\mathbf{V}, \mathbf{E})$. It provides a method to process $\mathbf{V}$ sequentially with a *deterministic* ordering. The algorithm of topological batching is outlined in the extended version. In brief, nodes are organized by a topological sort, with each $\mathbf{B}_i$ representing a topological equivalence class of $\mathbf{V}$.

Topological batching was initially introduced in previous work (Crouse et al. 2019) and refined with rigorous mathematical proof by Thost (Thost and Chen 2021). Since we assume a causal graph is a DAG, topological batching can be directly applied to SCMs. See Figure 4a for an illustration. In this example, the label set is partitioned into an ordered sequence $\mathbf{B} = (\{1, 2\}, \{3\}, \{4\})$.

4 Causally Consistent Normalizing Flows

Definitions

Sequential representation of an SCM. Similar to the graph representation, *The sequential representation of an SCM* entails describing the SCM by an ordered sequence. Specifically, for a causal graph $\mathcal{G}$ which is a DAG, we can obtain an ordered sequence $\mathbf{B} = (\mathbf{B}_1, \cdots, \mathbf{B}_n)$ by applying topological batching on $\mathcal{G}$ as discussed in Section 3. The sequence $\mathbf{B}$ could be interpreted as a sequential representation of the SCM. For instance, the sequential representation of Figure 4a is $(\{1, 2\}, \{3\}, \{4\})$.

Partial causal transformation. Assume we have two random variables: $\mathbf{U} = \{U_1, \cdots, U_d\}$, $\mathbf{X} = \{X_1, \cdots, X_d\}$, a label subset $\mathbf{L} \subseteq \{1, \cdots, d\}$ and the parent node label set $\mathbf{pa}_i$ of each X_i as defined in Section 3. A *partial causal transformation* T_θ over the label set $\mathbf{L}$ can be expressed as follows:

$$X_i = \begin{cases} T_\theta(U_i \mid \mathbf{U}_{\mathbf{pa}_i}) & \forall i \in \mathbf{L} \\ U_i & \forall i \notin \mathbf{L} \end{cases} \tag{6}$$

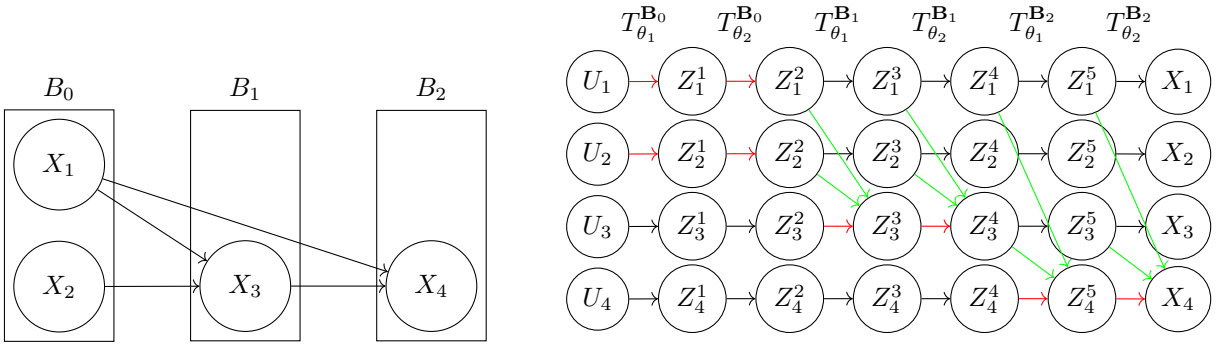
We call it "partial" because it only transfers U_i to X_i under the condition $\mathbf{U}_{\mathbf{pa}_i}$ for any $i \in \mathbf{L}$. In the following section, we use $T_\theta^{\mathbf{L}}$ to express the partial causal transformation T_θ over the label set $\mathbf{L}$.

Causally Consistent Normalizing Flows. Given the sequential representation $\mathbf{B} = (\mathbf{B}_1, \cdots, \mathbf{B}_n)$ of an SCM $\mathcal{M}$, we define *Causally Consistent Normalizing Flows* as such NF whose transformation is $T_\theta^{\mathbf{B}} = T_{\theta_n}^{\mathbf{B}_n} \circ \cdots \circ T_{\theta_1}^{\mathbf{B}_1}$.

For convenience, we use $\mathbf{Z} = (\mathbf{Z}^0 = \mathbf{U}, \cdots, \mathbf{Z}^{n-1}, \mathbf{Z}^n = \mathbf{X})$ to represent the output of each NF in the chain. More specifically, we use Z_i^k to represent the i-th output of $\mathbf{Z}^k$.

A Running Example

We implement CCNF based on the SCM illustrated in Figure 4a as an example. Recall that the sequential representation of Figure 4a is $\mathbf{B} = (\mathbf{B}_0, \mathbf{B}_1, \mathbf{B}_2) = (\{1, 2\}, \{3\}, \{4\})$.



(a) An SCM and its related topological batching result

(b) An example of CCNF based on Figure 4a with color. See Section 4 for details.

Figure 4: An example SCM, its topological order, and a related CCNF.

The CCNF comprises two layers for each $T_{\theta_i}^{\mathbf{B}_i}$ and is depicted in Figure 4b. Note that there is no limitation on the number of layers for $T_{\theta_i}^{\mathbf{B}_i}$, and we pick two for convenience.

The details of the graph in Figure 4b are elaborated below. Each column represents $(\mathbf{Z}^0, \cdots, \mathbf{Z}^n)$ as previously described. We use right arrows with different colors to denote the direction of the dataflow. The black arrow indicates the left node is equivalent to the right node, such as $U_3 \rightarrow Z_3^1$ meaning $U_3 == Z_3^1$. The green arrow indicates that the value of the left node is utilized during training to determine the parameters. For instance, the green arrows between $Z_1^2 \rightarrow Z_3^3$ and $Z_2^2 \rightarrow Z_3^3$ suggest that Z_1^2 and Z_2^2 are employed to determine the parameter θ_1. The red arrow indicates the left variable is the dependent variable on the right variable. For instance, $Z_3^2 \rightarrow Z_3^3$ denotes $Z_3^3 = T_{\theta_1}^{\mathbf{B}_1}(Z_3^3)$. Specifically, since there is no dependence for the variables $X_1, X_2 \in \mathbf{B}_0$, the $T_{\theta_0}^{\mathbf{B}_0}$ could be emitted and we can obtain the value of X_1, X_2 by directly sampling from the distribution.

5 CCNF: Properties and Operations

Properties of CCNF

Theorem 5.1 (Causality)**.** *Given a CCNF $T_\theta^{\mathbf{B}}$, for the i-th variable X_i, X_i only depends on its parents $\mathbf{X}_{\mathbf{pa}_i}$ and U_i. Particularly assume $i \in \mathbf{B}_j$, we have $X_i = T_{\theta_j}^{\mathbf{B}_j}(U_i \mid \mathbf{X}_{\mathbf{pa}_i})$*

Theorem 5.2 (Universality)**.** *A CCNF $T_\theta^{\mathbf{B}}$ is a multi-layer universal approximator as long as for any j, $T_{\theta_j}^{\mathbf{B}_j}$ is a multi-layer universal approximator.*

Theorem 5.3 (Causal Consistency)**.** *$T_\theta^{\mathbf{B}}$ is causally consistent with the given SCM $\mathcal{M}$.*

Theorem 5.4 (Minimum Layer)**.** *If the longest path of the DAG causal graph $\mathcal{G}$ is d, then CCNF contains at least d layers.*

Proofs of the theorems are available in the extended version. All those properties together make CCNF highly practical. Causality guarantees that all causal relationships within the SCM are encapsulated. Universality guarantees CCNF can approximate the distributions of endogenous random variables in any form. Causal consistency and minimum layer limitation collectively guarantee CCNF can capture accurate causality in $\mathcal{M}$.

Causal Inference Tasks

According to Pearl's causal hierarchy, causal inference tasks can be divided into three levels: observations, interventions, and counterfactuals. Here we will demonstrate that CCNF can perform all three tasks effectively. The extended version contains algorithms for all three tasks.

Observations. Generating observation data in CCNF is straightforward. We begin by sampling $\mathbf{U}$ from the given distributions $P_{\mathbf{U}}$. Then we compute $\mathbf{X}$ through $\mathbf{X} = T_\theta^{\mathbf{B}}(\mathbf{U})$. We repeat this process and get multiple possible $\mathbf{X}$s to form the observations $\mathbf{O}$.

The Do Operator. Before diving into interventions and counterfactuals, it is crucial to introduce the *do operator* (Pearl 2012), since it forms the foundation for those concepts. $Do(X_i = a)$ simulates a physical intervention on an SCM $\mathcal{M} = (\tilde{\mathbf{f}}, P_{\mathrm{u}})$ by fixing the observational variable X_i to a specific value a. Traditionally, the do operator requires modifying the $\mathcal{M}$ by substituting the variable X_i from a in every function $\tilde{\mathbf{f}}$. However, this approach is not suitable for CCNF. Instead, we propose a method like Causal NF to address this limitation. The key insight lies on the fact that $X_i = T_{\theta_j}^{\mathbf{B}_j}(U_i \mid \mathbf{X}_{\mathbf{pa}_i})$, indicating that fixing the value of X_i is equivalent to fixing the value of U_i.

Interventions. Interventions can be realized as applications of the do operator. More particularly, given $\mathbf{X} = (X_1, \cdots, X_d)$ with d attributes, interventions inquire about the distributions of variables when X_i is fixed to a, i.e. $P(X_j \mid X_i = a), j \in \{1, \ldots, d\}$. In practice, we generate observations $\mathbf{O}$ as described before. For each $\mathbf{X} \in \mathbf{O}$, we constraint the value of X_i in $\mathbf{X}$ by performing $Do(X_i = a)$ on $\mathbf{X}$. The constrained results represent samples from the distributions $P(\mathbf{X} \mid Do(X_i = a))$.

Counterfactuals. Like interventions, counterfactuals can also be accomplished through the do operator. Specifically, counterfactuals seek the precise value of $X_j, j \in \{1, \ldots, d\}$

when the set X_i is fixed to its counterfactual, i.e. $X_i \leftarrow X_i^{cf}$. In practice, For any given $\mathbf{X}$, we execute $Do(X_i = X_i^{cf})$ on it to get the counterfactual of $\mathbf{X}$.

6 Evaluation

We evaluate CCNF to answer three key questions:

1. **Causal Consistency.** Despite theoretical assurances of causal consistency is demonstrated, does CCNF maintain this consistency in practical implementations?
2. **Performance on Causal Inference Tasks.** In causal inference tasks, How accurately do the data instances generated by CCNF compare with those generated by actual models? Is there an observable improvement in accuracy compared to state-of-the-art models?
3. **Effectiveness in Real-world Case Studies.** Can CCNF be effectively applied to real-world scenarios, such as mitigating unfairness?

Refer to the links of the extended version and code for a complete description of the experiments.

Causal Consistency

Experiment design. We compare CCNF with three state-of-the-art models: CAREFL, VACA, and CausalNF. CAREFL is the first causal autoregressive flow, utilizing causal ordering with an affine layer to capture casualty. VACA employs a GNN to encode the causal graph, leveraging its structure to capture causal relationships. CausalNF restricts the conditioner of autoregressive flow to the parent nodes, enhancing its ability to model causal dependencies. All models except CCNF are categorized into two types: a model with one layer ($L = 1$) and a model with more than one layer ($L > 1$). More details are in the extended version.

Measurement. In this experiment, we evaluate given models based on two key metrics: accuracy and causal consistency. Accuracy reflects through the KL distance between the captured prior distribution and the actual one, denoted as $KL(p_{\mathcal{M}}|p_\theta)$. Causal consistency is reflected through Equation 7 as in Causal NF.

$$\mathcal{L}(T_\theta(X)) = \|\nabla_x T_\theta(X) \cdot (1 - G)\|_2 \tag{7}$$

Here, G represents the causal graph as an adjacency matrix of a given SCM $\mathbf{M}$, $\nabla_x T_\theta(X)$ denotes the Jacobian matrix of $T_\theta(X)$. $T_\theta(X)$ is causally consistent with $\mathbf{M}$ iff $\mathcal{L}(T_\theta(X)) = 0$.

Result. Results are summarized in Table 2. In a nutshell, the practical results are consistent with theoretical expectations. Firstly, CCNF demonstrates causally consistent with the given SCM, as $\mathcal{L}(T_\theta(X))$ of CCNF is consistently 0. Secondly, state-of-the-art models can only keep consistency by constraining their expressive power. CAREFL fails to meet the casual consistency altogether, while VACA and CauslNF can only achieve that with a single layer ($L = 1$). These findings remain consistent with the results reported in their respective papers.

Causal Inference Tasks

Experiment design. Since only CausalNF and VACA with one layer (L = 1) can ensure causal consistency, in this experiment, we compare CCNF with CausalNF (L = 1) and VACA (L = 1) for causal inference tasks. We test given models on representative synthetic datasets: Nonlinear Triangle dataset, Nonlinear Simpson dataset, M-graph dataset, Network dataset, Backdoor dataset, and Chain dataset with 3–8 nodes, respectively. Those causal structures are either from previous works (Javaloy, Martin, and Valera 2023; Sánchez-Martin, Rateike, and Valera 2022) or from practical applications, making them suitable for evaluating the performance of causal inference.

Measurement. We use three different measurements to evaluate the performance of causal inference tasks. For observations, the measurement is the KL distance, for interventions, we measure the max *Maximum Mean Discrepancy* (MMD) distance. For counterfactuals, we measure the *Root-Mean-Square Deviation* (RMSD) distance. More details are in the extended version.

Result. As shown in Table 3, CCNF demonstrates superior performance compared with previous works across nearly all datasets. This can be attributed to the ability of CCNF to capture complex casualties with additional middle layers—an ability absent in CausalNF and VACA. While compared with CausalNF, CCNF requires approximately 130% more time for training and evaluation, the time spent remains within a reasonable range and is less than that of VACA. Overall, CCNF is a more practical choice for causal inference tasks compared to stat-of-the-art tools.

Real-world Evaluation

Experiment design. Like prior works (Sánchez-Martin, Rateike, and Valera 2022; Javaloy, Martin, and Valera 2023), we select the German credit dataset as a representative example. Classifiers commonly utilize this dataset to predict the credit risk for a given applicant. If a classifier of the German credit predicts the risk directly through the *sex* attribute, we say it has the *unfairness problem*.

CCNF offers two methods to address real-world issues: ① building a fairness classifier directly or strengthening the dataset through counterfactual data augmentation. Specifically, we first train CCNF on the German credit dataset. For any applicant, CCNF could function as an unfairness-free classifier by setting the exogenous variable of the risk attribute to its mean value, which is 0 in our experiment. Additionally, ② CCNF could generate counterfactuals of the training set to create a data-augmented dataset. We also build an SVM classifier on the origin German credit dataset and the augmented one respectively for comparison.

Measurement. We utilize the *individual fairness* (Fleisher 2021) to evaluate the fairness of a classifier. Individual fairness is determined by examining whether changing the sex attribute of an applicant alters the risk level predicted by a classifier. Mathematically, for a test dataset with n instances, we define *individual fairness* as $\sum_{i=1}^{n} |Risk(X_{sex=1}) - Risk(X_{sex=0})|/n$.

	L = 1			L > 1			
	CAREFL	**CausalNF**	**VACA**	CAREFL	CausalNF	VACA	**CCNF**
KL	0.01±0.03	0.00±0.00	2.96±0.08	0.00±0.00	0.00±0.00	2.62±0.08	0.00±0.00
$\mathcal{L}(T_\theta(X))$	0.20±0.04	**0.00±0.00**	**0.00±0.00**	0.32±0.09	0.16±0.05	0.15±0.01	**0.00±0.00**

Table 2: Causal consistency comparison between CCNF and prior works. The causally consistent models are marked in bold. KL is used to evaluate the accuracy and $\mathcal{L}(T_\theta(X))$ is used to evaluate the causal consistency

Dataset	Model	Performance			Time(ms)	
		KL	Inter.$_{\text{MMD}}$	C.F.$_{\text{RMSD}}$	Train	Evaluation
Nlin-Triangle	CCNF	**0.12±0.04**	**0.03±0.03**	**0.14±0.05**	20.24±0.32	19.80±0.27
	CausalNF	0.37±0.00	0.10±0.02	0.79±0.07	**6.03±0.07**	**5.09±0.10**
	VACA	1.41±0.07	2.13±0.61	34.62±12.53	36.23±0.70	35.27±0.63
Nlin-Simpson	CCNF	**0.01±0.00**	**0.00±0.00**	**0.00±0.00**	15.96±0.24	15.73±0.68
	CausalNF	0.25±0.00	0.04±0.01	**0.00±0.00**	**6.30±0.13**	**5.53±0.30**
	VACA	1.56±0.04	0.11±0.15	0.59±0.10	36.52±0.40	35.62±0.39
M-Graph	CCNF	**0.01±0.00**	**0.00±0.00**	0.04±0.01	9.13±0.11	8.36±0.18
	CausalNF	0.32±0.00	0.17±0.01	**0.02±0.01**	**6.21±0.15**	**5.33±0.26**
	VACA	1.83±0.01	0.12±0.01	1.19±0.03	40.50±1.18	39.89±0.72
Network	CCNF	**0.55±0.13**	**0.01±0.01**	**0.25±0.06**	19.94±0.32	19.44±0.86
	CausalNF	1.38±0.04	0.15±0.02	0.41±0.03	**9.11±0.08**	**8.14±0.07**
	VACA	1.67±0.03	0.38±0.12	13.20±0.50	38.36±0.36	37.59±0.36
Backdoor	CCNF	**0.56±0.00**	**0.01±0.00**	0.04±0.01	15.21±0.22	14.71±0.62
	CausalNF	0.96±0.00	0.08±0.02	**0.04±0.00**	**6.30±0.23**	**5.45±0.40**
	VACA	1.78±0.01	0.13±0.01	1.88±0.02	39.04±0.39	38.21±0.40
Chain	CCNF	**0.28±0.02**	**0.00±0.00**	**0.03±0.00**	26.10±0.34	24.78±0.29
	CausalNF	4.67±0.04	0.05±0.01	0.13±0.02	**11.59±0.09**	**10.75±0.07**
	VACA	1.52±0.14	0.22±0.03	1.26±0.26	45.69±0.72	44.66±0.66

Table 3: Causal inference tasks comparison between causally consistent models. In the title, *Inter.* means interventions, *C.F.* means counterfactuals. The best results are marked in bold

Name	Accuracy	F1	Fairness
SVM	73.00±0.00	82.12±0.00	9.00±0.00
SVM$_{\text{CF}}$	72.60±1.10	81.90±0.59	4.10±1.40
CCNF	**75.80±2.22**	**84.34±2.22**	**0.00±0.00**

Table 4: Real-world evaluation on German credit dataset, every number is magnified 100 times.

Result. The results are summarized in Table 4. Overall, CCNF can enhance fairness while maintaining accuracy in both methods. For the fairness problem, CCNF can lower the fairness rate significantly. Particularly, the NF classifier demonstrates superior accuracy and eliminates unfairness. Those facts reveal that CCNF are suitable for real-world problems like unfairness without compromising accuracy.

7 Conclusion and Future Work

Causal inconsistency in generative models can lead to significant consequences. While some prior works cannot guarantee causal consistency, others achieve it only by limiting their depth. In the paper, we introduce CCNF, a novel causal GM that ensures causal consistency without limiting the depth as rigorously demonstrated in Section 5. Furthermore, we elaborate on how to perform causal inference tasks in CCNF, demonstrating its proficiency in efficiency. Through synthetic experiments, we illustrate that CCNF outperforms state-of-the-art models in terms of accuracy across different causal tasks. To validate the real-world applicability of CCNF, we also apply CCNF to a real-world dataset and address practical issues concerning unfairness. Our results indicate that CCNF can effectively mitigate problems while maintaining accuracy and reliability. Overall, CCNF represents a significant advancement in incorporating generative models with causality, offering both theoretical guarantees of causal consistency and practical applicability in addressing real-world issues involved with causality.

Future work. First, a major constraint of CCNF is the requirement of a well-defined causal graph, which is challenging to obtain in reality. In the future, CCNF should be able to handle these cases more properly by supporting even incorrect or non-DAG causal graphs. Second, while CCNF primarily focuses on causal inference tasks, its capabilities could be extended to other domains. For instance, by incorporating appropriate causalities to the SCM, CCNF could aid in causal discovery tasks. In the future, CCNF could be leveraged for extensive tasks.

Acknowledgements

This work is funded in part by NSERC (RGPIN-2022-03325) and research gifts from Amazon and Meta.

References

Crouse, M.; Abdelaziz, I.; Cornelio, C.; Thost, V.; Wu, L.; Forbus, K.; and Fokoue, A. 2019. Improving graph neural network representations of logical formulae with subgraph pooling. *arXiv preprint arXiv:1911.06904*.

Draxler, F.; Wahl, S.; Schnörr, C.; and Köthe, U. 2024. On the universality of coupling-based normalizing flows. *arXiv preprint arXiv:2402.06578*.

Fleisher, W. 2021. What's fair about individual fairness? In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, 480–490.

Harshvardhan, G.; Gourisaria, M. K.; Pandey, M.; and Rautaray, S. S. 2020. A comprehensive survey and analysis of generative models in machine learning. *Computer Science Review*, 38: 100285.

Hofmann, H. 1994. Statlog (German Credit Data). UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5NC77.

Javaloy, A.; Martin, P. S.; and Valera, I. 2023. Causal normalizing flows: from theory to practice. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Khemakhem, I.; Monti, R. P.; Leech, R.; and Hyvärinen, A. 2021. Causal Autoregressive Flows. arXiv:2011.02268.

Kocaoglu, M.; Snyder, C.; Dimakis, A. G.; and Vishwanath, S. 2017. CausalGAN: Learning Causal Implicit Generative Models with Adversarial Training. arXiv:1709.02023.

Komanduri, A.; Wu, X.; Wu, Y.; and Chen, F. 2024. From Identifiable Causal Representations to Controllable Counterfactual Generation: A Survey on Causal Generative Modeling. arXiv:2310.11011.

Papamakarios, G.; Nalisnick, E.; Rezende, D. J.; Mohamed, S.; and Lakshminarayanan, B. 2021. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57): 1–64.

Papamakarios, G.; Pavlakou, T.; and Murray, I. 2018. Masked Autoregressive Flow for Density Estimation. arXiv:1705.07057.

Pawlowski, N.; Coelho de Castro, D.; and Glocker, B. 2020. Deep structural causal models for tractable counterfactual inference. *Advances in neural information processing systems*, 33: 857–869.

Pearl, J. 2009. *Causality*. Cambridge university press.

Pearl, J. 2012. The do-calculus revisited. *arXiv preprint arXiv:1210.4852*.

Ribeiro, F. D. S.; Xia, T.; Monteiro, M.; Pawlowski, N.; and Glocker, B. 2023. High Fidelity Image Counterfactuals with Probabilistic Causal Models. arXiv:2306.15764.

Sanchez, P.; and Tsaftaris, S. A. 2022. Diffusion causal models for counterfactual estimation. *arXiv preprint arXiv:2202.10166*.

Sánchez-Martin, P.; Rateike, M.; and Valera, I. 2022. VACA: Designing variational graph autoencoders for causal queries. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 8159–8168.

Thompso, E. 2023. Class-action lawsuit against Facebook claiming discrimination gets the green light. *CBC*.

Thost, V.; and Chen, J. 2021. Directed Acyclic Graph Neural Networks. arXiv:2101.07965.

Xi, Q.; and Bloem-Reddy, B. 2023. Indeterminacy in generative models: Characterization and strong identifiability. In *International Conference on Artificial Intelligence and Statistics*, 6912–6939. PMLR.

Xia, K.; Pan, Y.; and Bareinboim, E. 2022. Neural causal models for counterfactual identification and estimation. *arXiv preprint arXiv:2210.00035*.

Yang, M.; Liu, F.; Chen, Z.; Shen, X.; Hao, J.; and Wang, J. 2021. Causalvae: Disentangled representation learning via neural structural causal models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9593–9602.