NeuroKoopman Dynamic Causal Discovery

Rahmat Adesunkanmi, *Student member, IEEE*, Balaji Sesha Srikanth Pokuri,
Ratnesh Kumar, *Fellow, IEEE*

arXiv:2404.16326v1 [cs.LG] 25 Apr 2024

Abstract—In many real-world applications where the system dynamics has an underlying interdependency among its variables (such as power grid, economics, neuroscience, omics networks, environmental ecosystems, and others), one is often interested in knowing whether the past values of one time series influences the future of another, known as Granger causality, and the associated underlying dynamics. This paper introduces a Koopman-inspired framework that leverages neural networks for data-driven learning of the Koopman bases, termed NeuroKoopman Dynamic Causal Discovery (NKDCD), for reliably inferring the Granger causality along with the underlying nonlinear dynamics. NKDCD employs an autoencoder architecture that lifts the nonlinear dynamics to a higher dimension using data-learned bases, where the lifted time series can be reliably modeled linearly. The lifting function, the linear Granger causality lag matrices, and the projection function (from lifted space to base space) are all represented as multilayer perceptrons and are all learned simultaneously in one go. NKDCD also utilizes sparsity-inducing penalties on the weights of the lag matrices, encouraging the model to select only the needed causal dependencies within the data. Through extensive testing on practically applicable datasets, it is shown that the NKDCD outperforms the existing nonlinear Granger causality discovery approaches.

Index Terms—Granger causality, Time series, Koopman operator, Nonlinear dynamics

1 Introduction

1.1 Motivation and Related Works

CAUSALITY analysis is vital in multivariate time series study for identifying potential cause and effect-informed networked dynamic relationships present in the observational data [1]. Granger causality (GC) corresponds to the situation in which the past of one time series helps predict the future of another [2], [3]. The notion of GC has been utilized in many fields of networked dynamics, including finance [4], neuroscience [5], meteorology [6], economics [7]. In neuroscience [8], [9] and biology [10], [11], for example, such analyses help determine whether a past activity in one brain region influences a later activity in another region, and to infer the underlying gene regulatory networks, respectively.

Many traditional methods for estimating GC assume that the time series follows linear dynamics and thus use a vector autoregressive (VAR) model [3], [11], where for a vector of n variables at time t, $x_t = (x_t(1), \ldots, x_t(n))$, the following linear regression holds:

$$x_t = \sum_{l=1}^{L} W_l x_{t-l} + e_t. \tag{1}$$

Here $W_l, l = 1, \ldots, L$ are lag matrices, L is the maximum lag ("memory of the system"), and e_t is additive noise, typically assumed to be a zero mean white noise. GC dependency then corresponds to the non-zero entries in $W_l, l \in [1, L]$. Authors in [12] later gave an equivalent definition of GC based on coefficients in a moving average (MA) representation (see [13]). GC for an $(i,j) \in [1,n]^2$ pair can be tested statistically using an F-test [14] comparing two models: an estimated model, which includes past values of both $x(i)$ and $x(j)$ versus another estimated model, that includes the past values of only $x(i)$. $x(j)$ is declared Granger causal for $x(i)$ if the F-test hypothesis of $x(i)$'s dependence on $x(j)$ is accepted.

Sparsity-inducing regularizers, such as lasso [15] or group lasso [16], are used to identify the minimally required number of causal dependencies [11]. These regularizers result in only a limited number of GC connections for each time series. Additionally, it is necessary to define the maximum time delay to be considered in GC analysis. To address the issue of choosing the relevant number of lags without causing overfitting, methodologies such as the hierarchical [17] and truncating lassos [18] have been proposed.

When data dynamics are nonlinear, imposing linear models can lead to inconsistent estimation of the true Granger causal interactions [19], [20]. Model-free methods like transfer entropy [21] or directed information [22] avoid linearity assumption, but only infer connectivity information and not the underlying dynamics [23]. Deep Learning (DL) and ML methods have shown promise in learning the complex dynamics of a system. Neural networks (NNs) and their variants, such as autoregressive multilayer perceptrons [24], [25] (MLPs) and long-short term memory networks (LSTMs) [26], are capable of modeling nonlinear interactions [27]. However, these methods may lack interpretability if not carefully modeled. Accordingly, [28] introduced component-wise MLPs and LSTMs, termed cMLP and cLSTM, that support the interpretability of GC from the learned models. Search for parameters of these models is also coupled with sparsity-inducing penalties.

In this research, we propose our approach of inferring causal structure and underlying dynamics using a Koopman-inspired lifting of the given time series data into a higher-dimensional space where the evolution appears

This work was supported in part by U.S. National Science Foundation under grant NSF-CSSI-2004766 and NSF-PFI-2141084.

Rahmat Adesunkanmi, Balaji Sesha Srikanth Pokuri, Ph.D. students, and Ratnesh Kumar, Palmer Professor, are with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50010, USA (e-mail: rahma, bspokuri, rkumar@iastate.edu.)
(Corresponding author: Rahmat Adesunkanmi.)

This work has been submitted to the IEEE for possible publication. Copyright may be transferred without notice, after which this version may no longer be accessible.

linear reliably; use data-driven auto-learning of the lifting and projection functions in the form of neural networks; and next infer a sparse linear regressive model in the lifted domain. The traditional approach to Koopman linear embedding employs a large number of nonlinear basis functions for approximating nonlinear systems into higher dimensional linear representations [29] and for projecting the lifted linear dynamics back to the base space [30]. However, to select the basis functions for lifting/projection, the traditional methods rely on ad hoc predefined dictionaries of bases, such as polynomial or radial functions, which may not be optimal. Some ways to obtain approximations of the Koopman operators involve utilizing the methods of dynamic mode decomposition (DMD) [31] and the extended-DMD (EDMD) [30], [32], where again, one assumes the knowledge of a predefined dictionary of basis functions. Inferring causal interactions, utilizing the Koopman embedding, is a relatively young field. Authors in [33] infer causal interaction in a dynamical system by using transfer Perron-Frobenius (P-F) and Koopman operators, whereas [34] utilized the setting of DMD towards GC inference.

In contrast to the traditional predefined basis dictionary, the data-driven approach provides a potential to learn basis functions accurately, without needing any user-designated bases [35], thereby enabling a scalable auto-encoding of the basis functions. NNs have been used to learn Koopman embeddings [36], [37]. Recently, we have also demonstrated their application to model-predictive control (MPC) of smart grids by employing auto-learned Koopman bases to accelerate MPC computation [38]. In this research, we learn the GC relation and dynamics via a linear regression applied to the nonlinearly transformed time series, employing a data-driven NN-based autoencoder for the first time:

$$\varphi(x_t) = \sum_{l=1}^{L} W_l \varphi(x_{t-l}) + e_t, \tag{2}$$

where the intrinsic nonlinear lifting map φ, the causal interactions among the lifted variables captured by lag matrices $W_l, l \in [1, L]$, as well as their projection map $\varphi^{-1}(\cdot)$, are all learned simultaneously in one go from the observed time series data. (The dimension of the auto-encoded lifted space is a hyperparameter and is chosen by exploration.)

1.2 Our Approach and Key Contributions

We summarize here the key contributions of our framework "NeuroKoopman Dynamic Causal Discovery (NKDCD)" with the following key contributions:

- We introduce the NKDCD architecture for causal discovery and learning underlying dynamics, comprising of:
  - A data-driven basis-dictionary free approach to learning the NN-based bases for linear Koopman embedding required of capturing sparse nonlinear dependency in from a higher-dimensional sparse linear model (5);
  - A NN-based data-driven learning of sparse linear model in the lifted domain to infer GC and underlying dynamics;
  - A data-driven learning of projection function to map down both the lifted domain time series and the estimated high-dimensional sparse linear models to the base space.
- The linear embedding approach is basis dictionary-free, in which the bases are auto-learned from the data, unlike the traditional approaches that manually pick basis functions that generally do not lead to optimal solutions.
- Each element of the time series is lifted identically to its higher-dimensional representation, $x(i) \in \mathbb{R}^1 \mapsto \varphi(x(i)) \in \mathbb{R}^{1\times N}, i \in [1, n]$, thereby preserving variable-wise separation at the lifted level, to ensure variable-wise disentanglement of the signals and their interdependencies in the lifted domain.
- For preserving the structure of sparsity and interdependencies of the original variables, group-level sparsity-inducing penalties are imposed in the lifted high-dimensional setting, where all lifted variables corresponding to a single variable of the original domain are placed in the same group.
- The autoencoder-based architecture is trained end-to-end in one go, learning directly from data the unknown basis function, the lag matrices associated with the lifted domain model, and the projection function together while minimizing the least-square prediction error and maximizing the sparsity of interdependencies. Such an end-to-end learning also eliminates the risks of numerical issues associated with iterative learning of the individual components (lifting, lag dynamics causality, and projection) [38].
- For accuracy and reliability, the loss function involves multiple terms to individually account for the accuracy of each NN block's input-output relationship, making the overall autoencoder architecture robust. These correspond to the accuracy of the temporal dependencies, the GC constraints, and the lifting and projection functions.
- Since appropriate lag selection is crucial for GC and underlying dynamics discovery, we impose additional structured group penalties that infer both the GC and the lags for each of the inferred interactions. Our formulation automatically selects a subset of the time series for each output time series that Granger causes it, regardless of the lag of the interaction.
- Our method is validated via simulations on a variety of datasets: a finance dataset [39], a nonlinear Lorenz-96 model dataset [40], brain-imaging fMRI dataset [41], and the DREAM3 gene regulatory network benchmark dataset [42]. Our results are able to outperform the ones reported in the literature.

2 Mathematical Framework

As outlined in [2], the notion of Granger causal interaction is formulated as follows: A time series y Granger causes time series x if the variance of the prediction error when the history of y is included is lower than the variance of the prediction error when the history of y is excluded, i.e.,

$$\sigma^2(x_t|\mathcal{H}_{<t}) < \sigma^2(x_t|\mathcal{H}_{<t}\backslash y_{<t}), \tag{3}$$

where $\mathcal{H}_{<t}$ represents the history containing all relevant information prior to time t, $\mathcal{H}_{<t}\backslash y_{<t}$ indicates the exclusion of $y_{<t}$ values from $\mathcal{H}_{<t}$, and $\sigma^2(\cdot)$ denotes the variance. Essentially, y is causal for x if past values of y improve the prediction of x.

2.1 Traditional VAR models for GC

Considering the model of (1), a time series element $x(j)$ Granger-causes another time series element $x(i)$ if there exists a value at any lag l for which $W_l(i,j) \neq 0$. GC analysis thus amounts to estimating the lag matrices from the time series observations, and determining the non-zero entries. The loss function includes the square of the prediction error together with lasso penalty [16], [43] on the lag matrices to attain their sparsity:

$$\min_{W_1,\ldots,W_L} \sum_{t=L+1}^{T} \|x_t - \sum_{l=1}^{L} W_l x_{t-l}\|_2^2 + \lambda\Omega(W_1,\ldots,W_L), \tag{4}$$

where $\|\cdot\|_2$ denotes the L_2 norm, and $\lambda > 0$ is a hyper-parameter that controls the weight ascribed to the sparsity-inducing lasso term $\Omega(W_1,\ldots,W_L)$ in the loss function. In our work, we explore multiple types of group lasso loss terms for inducing sparsity as described in (10)-(12).

2.2 Nonlinear autoregressive (NAR) model

The VAR model assumes linear dependence, which might not be the case for a practically applicable setting, in which case, the model may not yield the GC dependencies correctly. In practice, $x_t \in \mathbb{R}^n$ can evolve according to more general nonlinear dynamics. A general element-wise nonlinear autoregressive (NAR) model is of the following form [44]:

$$x_t(i) = f_i\Big(x_{t-1:t-L}(1),\ldots,x_{t-1:t-L}(n)\Big) + e_t(i), \tag{5}$$

where $x_{t-1:t-L}(j), j \in [1,n]$ is an abbreviation to denote the past L values, $x_{t-1}(j),\ldots,x_{t-L}(j)$, and f_i is a function that specifies how the past lags are mapped to $x_t(i)$. Granger non-causality of $x(j)$ for $x(i)$ corresponds to the situation when f_i does not depend on $x_{t-1:t-L}(j)$.

2.3 Proposed framework of NeuroKoopman Dynamic Causal Discovery (NKDCD)

Suppose we have an NAR time series data $\mathbf{x} = [x_1,\ldots,x_T] \in \mathbb{R}^{n\times T}$. We first lift each element of vector data at each time point to higher-dimensional intrinsic coordinates:

$$X_t^{koop}(i) = \varphi(x_t(i)) \tag{6}$$
$$\varphi \in \mathbb{R}^{1\times N}; X_t^{koop}(i) \in \mathbb{R}^{1\times N}(N >>> 1); i \in [1,n].$$

The lifted data is then modeled following VAR as in (2):

$$\widehat{X}_t^{koop} = \sum_{l=1}^{L} W_l X_{t-l}^{koop} \tag{7}$$
$$W_l \in \mathbb{R}^{(n\times N)\times(n\times N)}, l \in [1,L]; W_l(i,j) \in \mathbb{R}^{N\times N}, i,j \in [1,n],$$

where W_l's are the lag matrices in the lifted domain, and $\widehat{X}_t^{koop}, t \in [L+1,T]$, is the model estimate of X_t^{koop}. Both $\widehat{X}_t^{koop}$ and X_t^{koop} are projected element-wise using $\varphi^{-1} \in \mathbb{R}^{N\times 1}$ to the base space to yield:

$$\widehat{x}_t(i) = \varphi^{-1}(\widehat{X}_t^{koop}(i)) \qquad t \in [L+1,T] \tag{8}$$
$$\overline{x}_t(i) = \varphi^{-1}(X_t^{koop}(i)) \qquad t \in [1,T]. \tag{9}$$

Figure 1 depicts the NKDCD architecture and labels the base, lifted, and projected variables, the encoder that performs the lifting from the base space to a higher dimensional space, the decoder that performs the opposite, and the lag matrices for linear regression in the lifted domain.

Definition 1 (NKDCD Model-based GC)**.** *Referring to the NKDCD model (7), the time series $x(j)$ is Granger non-causal for time series $x(i)$ if for all $l \in [1,L], W_l(i,j) \in \mathbb{R}^{N\times N} = 0$, so the j^{th} time series does not contribute to the prediction of the i^{th} time series.*

The lifting function, φ, the projection function, φ^{-1}, and the lag matrices, $\{W_l, l \in [1,L]\}$, are learned from the observed time series data $\{x_1,\ldots,x_T\}$ in one go. To ensure learning robustness, we define a multi-term loss function, addressing the correctness of each component of the proposed NKDCD architecture.

1) *Correctness of mapping to lifted space and its projection to base space*: In the NKDCD framework, the dynamics evolve over the intrinsic coordinates $X^{koop} = \varphi(x)$ in the lifted domain, whereas the inverse $\overline{x} = \varphi^{-1}(X^{koop})$ recovers the base space value. For the accuracy of the lifting and projection pair, we introduce the following loss term:

$$\sum_{t=1}^{T} \|x_t - \varphi^{-1}(\varphi(x_t))\|^2.$$

2) *Correctness of linear VAR model in lifted space*: To discover the lag matrices that govern the GC and underlying dynamics, we learn the lag matrices $\{W_l, l \in [1,L]\}$ over the intrinsic coordinates, so that $\hat{X}_t^{koop} = \sum_{l=1}^{L} W_l X_{t-l}^{koop}$ holds. The accuracy of learning the lag matrices is achieved by introducing this loss term:

$$\sum_{t=L+1}^{T} \|\varphi(x_t) - \sum_{l=1}^{L} W_l \varphi(x_{t-l})\|^2.$$

3) *Correctness of NAR model in base space*: The projection operation allows us to map the linear VAR model of the lifted space to the NAR model of the base space. To achieve reconstruction accuracy of the NAR model, we introduce this additional loss term:

$$\sum_{t=L+1}^{T} \|x_t - \varphi^{-1}(\sum_{l=1}^{L} W_l \varphi(x_{t-l}))\|^2.$$

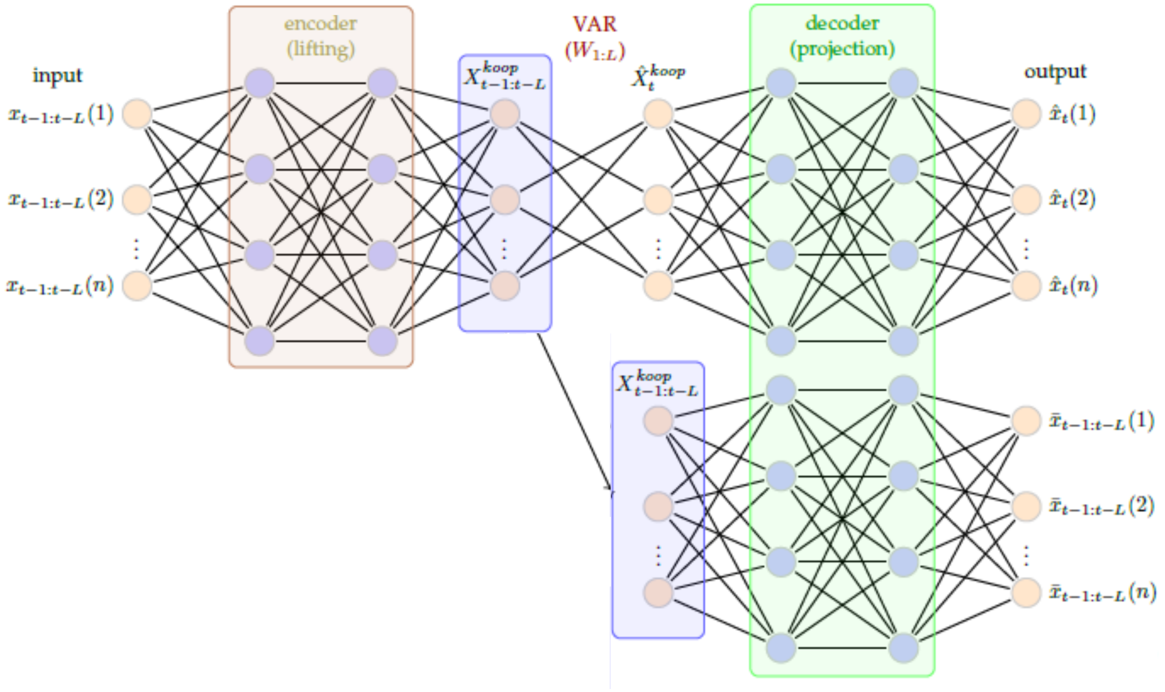


Fig. 1. Proposed NKDCD Architecture

4) *Correctness of NAR model with autoencoder*: The lifting and projection can be applied on the input side before comparing to the NAR output to enhance the end-to-end accuracy further, and for which we introduce this another loss term:

$$\sum_{t=L+1}^{T} \|\varphi^{-1}(\varphi(x_t)) - \varphi^{-1}(\sum_{l=1}^{L} W_l \varphi(x_{t-l}))\|^2.$$

5) *Sparsity of GC*: To have as few GC dependencies as needed, we add a fifth term to the loss function of the form $\lambda\Omega(W_1, ..., W_L)$, where $\lambda > 0$ is a tunable scalar hyperparameter, that penalizes having a large number of non-zero entries in $\{W_l, l \in [1, L]\}$. Since minimizing the count of non-zero entries is equivalent to minimizing the L_0-norm, which is discrete and hence discontinuous, one instead considers a relaxation to the L_1-norm. Since each $W_l(i, j), l \in [1, L]$ is a $N \times N$ matrix representing the lag-l weights between the N lifted components of the ith time series element and the N lifted components of the jth time series element, all such $N \times N$ elements of $W_l(i, j)$ for each $(i, j) \in [1, n]^2, l \in [1, L]$ are grouped together, thereby the sparsity penalty becomes an L_1-norm over groups, termed group lasso.

In addition, since all elements $W_l(i, j), l \in [1, L]$ must be zero to ensure that $x(i)$ is not GC dependent on $x(j)$, a grouping across the lags can be further employed. Grouping all the lags into a single group treats them equally, resulting in a "uniform-lag group lasso (ulg-lasso)":

$$\Omega(W_1, ..., W_L) = \sum_{ij} \|W_1(i, j), \ldots, W_L(i, j)\|_2. \quad (10)$$

The ulg-lasso of (10), however, runs the risk of treating the distant lag terms that may be absent in the ground truth the same as the near ones that are actually present in the ground truth. So a "hierarchical-lag group lasso (hlg-lasso)" that discounts more distant lags more severely is instead used sometimes:

$$\Omega(W_1, ..., W_L) = \sum_{ij} \sum_{l=1}^{L} \|W_l(i, j), \ldots, W_L(i, j)\|_2. \quad (11)$$

Finally, the most general case is when all the lags, near or distant, are treated independently, resulting in "independent-lag group lasso (ilg-lasso)":

$$\Omega(W_1, ..., W_L) = \sum_{ijl} \|W_l(i, j)\|_2. \quad (12)$$

Note as discussed above, the grouping in (12) corresponds to putting all $N \times N$ elements of $W_l(, i, j)$ into its own group for each $(i, j) \in [1, n]^2, l \in [1, L]$. Such grouping is also applicable to ulg- and hlg-lassos.

In summary, the overall loss function is defined as:

$$\begin{aligned}\mathcal{J} = & \sum_{t=1}^{T} \|x_t - \bar{x}_t\|^2 + \sum_{t=L+1}^{T} \|X^{koop}_t - \hat{X}^{koop}_t\|^2 \\ & + \sum_{t=L+1}^{T} \|x_t - \hat{x}_t\|^2 + \sum_{t=L+1}^{T} \|\bar{x}_t - \hat{x}_t\|^2 \\ & + \lambda\Omega(W_1, ..., W_L), \end{aligned} \quad (13)$$

where $\Omega(\cdot)$ takes the one of the forms in (10)-(12). For notational convenience, we denote the first four terms in $\mathcal{J}$ as $\mathcal{J}_1$ (which involves four L_2-norm terms, each of which is differentiable and so allows gradient descent for loss minimization) and denote the last term in $\mathcal{J}$ as $\mathcal{J}_2$, which being group lasso, involves inter-group L_1-norm and intra-group L_2-norm, that is not differentiable, requiring proximal-gradient descent for $\mathcal{J}_2$-loss minimization (in contrast, the $\mathcal{J}_1$-loss minimization uses the usual gradient descent).

In Section 3, we employ all three instances of group lasso for comparison and demonstrate how ilg-lasso outperforms hlg-lasso, which in turn surpassed ulg-lasso in performance.

2.4 Hyperparameters of NKDCD and loss function

In our autoencoder model of Figure 1, each time series is lifted to N dimensions, which is one of the hyperparameters. The maximum amount of lag is L, which is another hyperparameter. We have chosen the lifting and projection functions to have 2 hidden layers, where when the data has nonlinear dynamics, each hidden layer uses the leaky rectified linear unit (leakyReLU) as the activation function: $max\{0.1y, y\}$ and no activation function when the data has linear dynamics. The lifting hidden layer weight sizes are parameterized by h and are respectively, $1 \times \frac{h}{2}$, $\frac{h}{2} \times h$, and $h \times N$, while the projection hidden layer weight sizes are respectively, $N \times h$, $h \times \frac{h}{2}$, and $\frac{h}{2} \times 1$. The hyperparameter of the loss function is the sparsity weight factor λ. During the training of NKDCD, the learning rate (also called step-size) τ is yet another hyperparameter, and so is the threshold $\epsilon > 0$ used to view a weight parameter below the threshold as zero.

2.5 NKDCD optimization: (Proximal) gradient descent

Recall the decomposition $\mathcal{J}(= \mathcal{J}_1 + \mathcal{J}_2)$ of the loss function, in which the $\mathcal{J}_2$ portion of the loss function is only a function of the lag matrices $\{W_l, l \in [1, L]\}$, and it does not depend on the lifting and projection functions φ, φ^{-1}. As a result, the gradient of loss $\mathcal{J}$ with respect to φ, denoted $\Delta_\varphi(\mathcal{J})$, satisfies: $\Delta_\varphi(\mathcal{J}) = \Delta_\varphi(\mathcal{J}_1)$, and similarly, $\Delta_{\varphi^{-1}}(\mathcal{J}) = \Delta_{\varphi^{-1}}(\mathcal{L}_1)$. In contrast, the gradient of $\mathcal{J}$ vs. $\mathcal{J}_1$ with respect to the lag matrices are different from each other: $\Delta_{W_l}(\mathcal{J}) \neq \Delta_{W_l}(\mathcal{J}_1),\ l \in [1, L]$, because $\Delta_{W_l}(\mathcal{J}_2) \neq 0, l \in [1, L]$ (whereas in contrast, $\Delta_\varphi(\mathcal{J}_2) = \Delta_{\varphi^{-1}}(\mathcal{J}_2) = 0$).

Accordingly, in the mth iteration ($m \geq 0$) of update, the lifting and projection functions are updated using gradient descent with respect to only the respective $\mathcal{J}_1$ portions of the gradients, applied to the corresponding values from the mth iteration, as follows:

$$\varphi^{(m+1)} = \varphi^{(m)} - \tau\Delta_\varphi(\mathcal{J}_1)\Big|_{W_{1:L}^{(m)},\varphi^{(m)},(\varphi^{-1})^{(m)}}, \qquad (14)$$

$$(\varphi^{-1})^{(m+1)} = (\varphi^{-1})^{(m)} - \tau\Delta_{\varphi^{-1}}(\mathcal{J}_1)\Big|_{W_{1:L}^{(m)},\varphi^{(m)},(\varphi^{-1})^{(m)}}, \qquad (15)$$

in which $\tau > 0$ is the step size of the update (also called learning rate), and $(\cdot)^{(m)}$ denotes the value of $(\cdot)$ at an $m \geq 0$ iteration of the update. In contrast, the update for the lag matrices follows the following computation in the case of ulg-lasso, involving gradient descent for the $\mathcal{J}_1$ part of the loss (that is differentiable) and subsequently proximal gradient descent for the $\mathcal{J}_2$ part of the loss (that is not differentiable):

$$\widetilde{W}_l^{(m)} := W_l^{(m)} - \tau\Delta_{W_l}(\mathcal{J}_1)\Big|_{W_{1:L}^{(m)},\varphi^{(m)},(\varphi^{-1})^{(m)}} \qquad (16)$$

$$W_l^{(m+1)} = \text{prox}_{\tau\lambda\Omega}(\widetilde{W}_l^{(m)}), \qquad (17)$$

$$\text{prox}_{\tau\lambda\Omega}(\cdot)(i,j) = \left(1 - \frac{\tau\lambda}{||W_1^{(m)}(i,j),\ldots,W_L^{(m)}(i,j)||_2}\right)_+ (\cdot)(i,j), \qquad (18)$$

where the notation $(\cdot)_+ := \max\{0, (\cdot)\}$, and (18) computes the proximal gradient [45] for the case of ulg -group lasso. In the case of the hlg-lasso, the proximal gradient is computed iterating over a total of L rounds, where in the l^{th} round ($l \geq 1$), the subset $\{W_l^{(m,l)}, \ldots, W_L^{(m,l)}\}$ of round-l values are updated using the corresponding round-($l-1$) ones:

$$\begin{aligned}&\text{prox}_{\tau\lambda\Omega}(\cdot)(i,j)\\&= \left(1 - \frac{\tau\lambda}{||W_l^{(m,l-1)}(i,j),\ldots,W_L^{(m,l-1)}(i,j)||_2}\right)_+ (\cdot)(i,j),\end{aligned} \qquad (19)$$

where for the round-1 update ($l = 1$), the values used are:

$$W_l^{(m,0)} = \widetilde{W}_l^{(m)}, l \in [1, L].$$

Finally, the proximal gradient for ilg-lasso uses:

$$\text{prox}_{\tau\lambda\Omega}(\cdot)(i,j) = \left(1 - \frac{\tau\lambda}{||W_l^{(m)}(i,j)||_2}\right)_+ (\cdot)(i,j). \qquad (20)$$

It can be seen that $W_l^{(m+1)}, l \in [1, L], m \geq 0$ is computed from $W_l^{(m)}$ by first applying the gradient descent to obtain an intermediate value: $\widetilde{W}_l^{(m)} = W_l^{(m)} - \tau\Delta_{W_l}(\mathcal{J}_1)\Big|_{W_{1:L}^{(m)},\varphi^{(m)},(\varphi^{-1})^{(m)}}$, accounting for only the $\mathcal{J}_1$ portion of the loss (ignoring the $\mathcal{J}_2$ portion of the loss for the time being), and next, the value nearest to this intermediate value that also accounts for the $\mathcal{J}_2$ portion of the loss is found by applying the proximal gradient, $\text{prox}_{\tau\lambda\Omega}(\cdot)$, which for the ulg-lasso as well as ilg-lasso is given by a single round computation, employing Eqs. (18) and (20), respectively, and for the hlg-lasso it comprises of L-rounds of computations, where for each $l \in [1, L]$, the round-l computation is given by Eq. (19).

Considering the ulg-lasso penalty of (10) over the lag matrices $W_l(i, j), l \in [1, L]$, it shrinks each weight $W_l(i, j)$ equally, using (18). The hlg-lasso proximal gradient in (11), however, shrinks more those lag matrices that are further in the distant past, by using (19), which for each $l \geq 1$, applies l-rounds of shrinkage operations (so $W_1(i, j)$ undergoes only one round of shrinkage, whereas $W_L(i, j)$ witnesses L rounds of shrinkage). Finally, the shrinkage operation in the case of ilg-lasso (12) is independent for each lag using (20) and is also applied in a single round (like ulg-lasso). The resulting update steps are summarized in Algorithm 1.

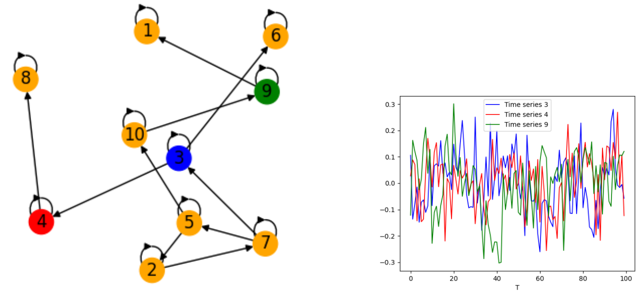
3 Comparing Group Lasso Penalties

To qualitatively visualize the sparsity pattern recovered under the three lasso penalties, we apply our framework to data generated from a sparse VAR(3) model with $n = 10$, presented in [28]. Here, each time series i depends on itself together with one other randomly selected time series

Algorithm 1 NKDCD

```
Require: time series data, $\mathbf{x} \in \mathbb{R}^{n \times T}$, $\lambda > 0$, $1 \geq \tau > 0$, $N >>> 1$, $L > 0$,
  Initialize: $m = 0, W_l^{(m)}, \varphi^{(m)}, (\varphi^{-1})^{(m)}$
  while not converged do
    compute $\{X^{koop}\}$ as in Eq. 6
    compute $\{\hat{X}^{koop}\}$ as in Eq. 7
    compute $\{\hat{x}\}$ as in Eq. 8
    compute $\{\bar{x}\}$ as in Eq. 9
    compute loss $\mathcal{T}$ as in Eq. 13
    Update $\varphi^{(m)}, (\varphi^{-1})^{(m)}, \{W_l^{(m)}, l \in [1, L]\}$ using (14) - (20) to get $\varphi^{(m+1)}, (\varphi^{-1})^{(m+1)}, \{W_l^{(m+1)}, l \in [1, L]\}$
    $m = m + 1$
  end while
  return $\varphi, \varphi^{-1}, \{W_l, l \in [1, L]\}$
```

among the remaining $n-1$ ones, and when time series i depends on time series j, $W_l(i,j) = 0.1$ for $l = 1, 2, 3$, while all the other lag entries are zero. Thus, this setting corresponds to the ground truth of $L = 3$. The ground truth causal graph and three selected time series are depicted in Figure 2. Time series 3 and 4 exhibit a causal relationship, while neither is causally linked to the time series 9. We collect $T = 1000$ sample points per time series for analysis.



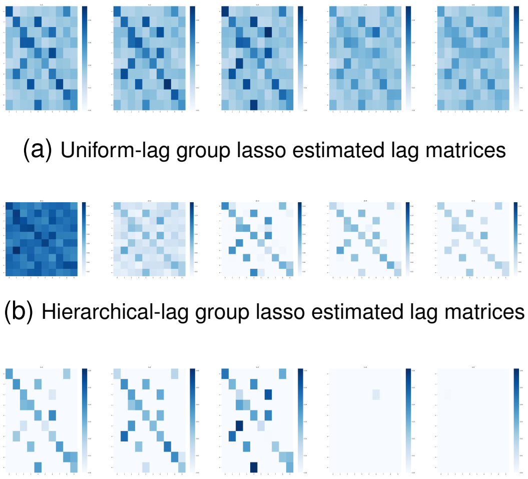
(a) Ground truth causal graph

(b) Observed time series

Fig. 2. VAR(3) data

Because the data already follows linear dynamics, the nonlinear activation in each hidden layer was omitted. For causal model discovery, we set the learning rate $\tau = 5 \times 10^{-2}$, the number of hidden layers in lifting and projection networks as 2 with $h = 4$, the scaling-up factor of each dimension during lifting as $N = 10$, batch size $= 500$, and the upper bound to lag $L = 5$. $\lambda = 2 \times 10^{-2}$ was chosen for the ulg- and ilg-lassos as an experimentally determined based penalty factor, while $\lambda = 10^{-4}$ was used for the hlg-lasso, which again is the corresponding best value determined experimentally. The entries of the lag matrices are shown as heat-maps in Figure 3, where each (i,j) entry is the norm of the $N \times N$ entries in the lifted-version $W_l(i,j)$ (all of which belong to a single group).

Taking a closer look at the captured blocks, i.e., for which $\|W_l(i,j)\| > \epsilon$ for a suitable threshold value of epsilon (chosen experimentally), it can be seen in the cases of ilg- and hlg-lassos that if $\|W_l(i,j)\|$ is shrunk to zero for some l, then $\|W_{l'}(i,j)\|, l' > l$ is also shrunk to zero. ilg- and hlg-lassos have different stopping epochs: 2500 vs 6200, which implies ilg-lasso converges faster than hlg-lasso. On the other hand, while ulg-lasso converges just as fast as ilg-lasso: 2500 iterations, it cannot accurately match the right value of L (as noted above, it runs the risk of treating the distant lag terms that may be absent in the ground truth the same as the near ones that are actually present in the ground truth). Based on these comparative observations of the three group lassos, moving forward, we have chosen to apply the NKDCD computations only for ilg- and hlg-lasso (both of which outperform ulg-lasso).



(a) Uniform-lag group lasso estimated lag matrices

(b) Hierarchical-lag group lasso estimated lag matrices

(c) Independent-lag group lasso estimated lag matrices

Fig. 3. Estimated 5-step lag matrices

4 Results for Practical Applications

Our study demonstrates the performance of our novel NKDCD approach in inferring Granger causality (GC) and underlying dynamics from both synthetic and practically applicable datasets. We conduct experiments on financial data [39], Lorenz-96 model data [40], fMRI brain-imaging model data [39], and DREAM3 genetic data [42], sourced from various existing studies.

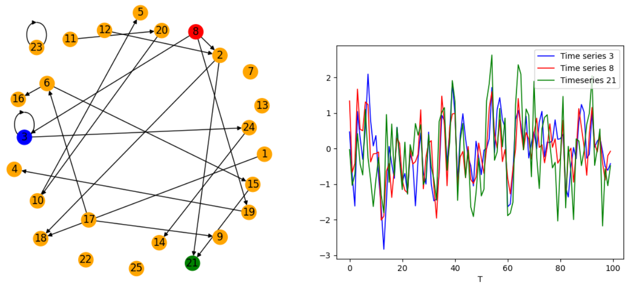
The results indicate that our NKDCD approach effectively discovers the underlying GC dependencies and nonlinear dynamics across various datasets. Key components of our model include a lifting encoder comprising multilayer perceptron (MLP) with 2 hidden layers, GC weight matrices, and a projection decoder, another MLP with 2 hidden layers. Each node in the encoder and decoder MLP has an activation function of a leakyReLU. We explore different hyperparameters, such as the learning rate (τ) varied over $[5 \times 10^{-4}, 5 \times 10^{-3}]$, and the sparsity weight (λ) varied over $[10^{-2}, 5 \times 10^{-2}, 3 \times 10^{-1}]$. See Table 1 for a complete list of the hyperparameters chosen experimentally that we use in analyzing the application datasets. We set a convergence criterion as the average loss per time series being less than the threshold of 0.9 and is no longer decreasing. The performance metrics are based on the area under the ROC curve (AUROC) and the area under the Precision-Recall curve (AUPR). We compare our results against linear VARs with hlg- and ilg-lasso for sparsity as formulated in Eqs. (4), (11), and (12) for baseline comparison, as well as the state-of-art results available in literature, including component-wise MLP (cMLP)/LSTM (cLSTM) as reported in [28], to demonstrate the gain resulting from the proposed NKDCD approach over the state-of-art. In our results, the weights corresponding to lag matrices, if they are small compared to a threshold $\epsilon > 0$ (decided case-by-case for optimal result), are viewed zero to infer GC. Our NKDCD approach shows promising results, outperforming the ones reported in the literature.

TABLE 1
Hyperparameters used for NKDCD Analysis

	Finance	Lorenz-96	fMRI	Dream3 (EC1,EC2,Y1)	Dream3 (Y2,Y3)
λ	.01	.05	.05	.01	.3
τ	.005	.0005	.0005	.0005	.005
L	3	5	3	2	2
N	15	15/45	10	5	10
h	4	16	8	8	8
batch	1024	500	500	966	966

4.1 Finance Data

The simulated finance dataset utilized in our study originates from research reported in [39] and is publicly available on GitHub. This dataset is fitted to the Fama-French Three-Factors Model [46], a framework that characterizes stock returns against three factors: volatility, size, and value. For our experiments, we specifically utilize dataset 20-1A, which comprises a 4000-day observation period across 25 stocks. In [39], the causal relationships among the time series present in this dataset evolve according to linear dynamics, and L is within 1-3. The ground truth interdependency is provided in Figure 4a and a sample of three selected time series elements is depicted in 4b, where the causal relationship between time series 3 and 8 is evident. At the same time, neither has a causal connection with the time series 21. In this analysis, we set the maximum lag (L) to 3, and since the data already follows linear dynamics, the nonlinear activation (namely, leakyReLU) is omitted within the hidden layers.



(a) Ground truth causal graph (b) Observed time series

Fig. 4. Finance data

The performance metrics are reported as the mean across five initializations with a 95% confidence interval. Results from our experiments on this finance dataset are summarized in Table 2. The AUROC and AUPR for the VAR model with the ulg- and hlg-lasso penalties are both 99.9 ± 0.1 and 97.6 ± 1.0, respectively. Our method performs just as well as the traditional VAR by giving AUROC and AUPR of 99.9 ± 0.1 and 98.0 ± 1.0 for the ilg-lasso penalty and 99.1 ± 0.0 and 90.9 ± 0.1 respectively for the hlg-lasso counterpart, showing that Koopman lifting preserves the causal relationship among the time series elements. These results demonstrate the effectiveness of our approach in uncovering the causal relationships and the underlying dynamics within the financial data having the 3-Factors linear model.

TABLE 2
Finance Data Results

Dataset	20-1A	
Metrics	AUROC	AUPR
NKDCD+Eq. 12	99.9 ± 0.1	98.0 ± 1.0
NKDCD+Eq. 11	99.1 ± 0.0	90.9 ± 0.1
VAR+Eq. 10	99.9 ± 0.1	97.6 ± 1.0
VAR+Eq. 11	99.9 ± 0.1	97.6 ± 1.0

4.2 Lorenz-96 Model with Nonlinear Dynamics

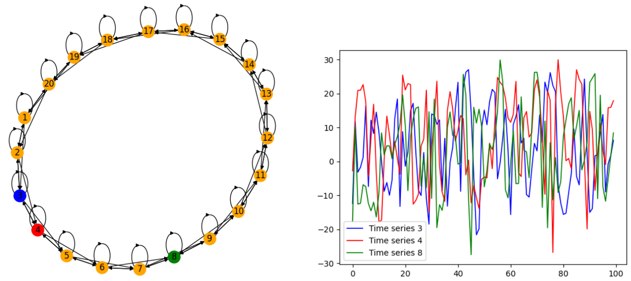
The Lorenz-96 model, introduced in [40], serves as a valuable tool for studying essential aspects of chaos theory, predictability, and the behavior of complex nonlinear interdependent dynamics. It provides a simplified yet insightful representation of the atmospheric dynamics, making it widely used in atmospheric science and related fields. The continuous dynamics in a Lorenz-96 model is defined for $i = 1, \dots, n, n \geq 4$ as follows:

$$\frac{dx_t(i)}{dt} = \Big(x_t(i+1) - x_t(i-2)\Big)x_t(i-1) - x_t(i) + F, \tag{21}$$

along with the following boundary correspondences:

$$x_t(-1) = x_t(n-1); x_t(0) = x_t(n); x_t(n+1) = x_t(1),$$

where F is a forcing constant that determines the level of chaos, and n is the dimension. For comparison with recent work, we utilized publicly available data of a Lorenz-96 model, described by Eq. 21 on GitHub, supplied by [28]. For our analysis, we chose $n = 20$ and a sampling rate of 0.1 for obtaining a discrete-time time series. The ground truth causal graph and three selected time series for $F = 40$ are depicted in Figure 5. Notably, time series 3 and 4 exhibit a causal relationship, while neither is causally linked to time series 8.



(a) Ground truth causal graph (b) Observed time series (selected)

Fig. 5. Lorenz-96 model data

We conduct simulations for $F = 10, 40$ and $T = 250, 500, 100$, as specified in Table 3, where our objective

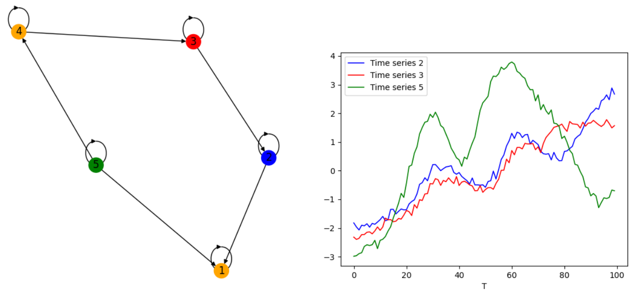
TABLE 3
Lorenz-96 Data AUROC Results as a function of Forcing Constant F and time series Length T

	$F = 10$			$F = 40$		
T	250	500	1000	250	500	1000
NKDCD+Eq. 12	94.1 ± 0.7	98.5 ± 0.2	99.8 ± 0.1	80.8 ± 0.1	92.0 ± 1.1	98.7 ± 0.1
NKDCD+Eq. 11	94.1 ± 0.6	98.5 ± 0.1	99.7 ± 0.1	82.6 ± 0.3	90.8 ± 0.7	96.7 ± 0.2
cMLP	86.6 ± 0.2	96.6 ± 0.2	98.4 ± 0.1	84.0 ± 0.5	89.6 ± 0.2	95.5 ± 0.3
cLSTM	81.3 ± 0.9	93.4 ± 0.7	96.0 ± 0.1	75.1 ± 0.9	87.8 ± 0.4	94.4 ± 0.5
VAR+Eq. 10	86.1 ± 0.6	86.2 ± 0.2	93.0 ± 0.4	70.0 ± 0.0	80.6 ± 0.0	84.7 ± 0.0
VAR+Eq. 11	86.1 ± 0.6	86.2 ± 0.2	93.0 ± 0.4	70.4 ± 0.0	81.1 ± 0.0	84.7 ± 0.0

is to compare the performance of the NKDCD approach against other state-of-art methods in literature, including component-wise MLP (cMLP)/LSTM (cLSTM) as reported in [28], and the standard linear VAR method. The performance metrics are reported as the mean across five initializations with a 95% confidence interval.

The results, summarized in Table 3, reveal a tradeoff between accuracy and length of observation as F increases. Despite challenges posed by shorter data lengths and higher chaos levels, NKDCD demonstrates superior performance over alternative methods in most scenarios, except for a single scenario when the data length is small ($T = 250$), and chaos is greater ($F = 40$), it is cMLP performs the best with NKDCD being the second best. At $F = 10, T = 250, 500, 1000$, the AUROCs for the VAR with ulg- and hlg-lassos are both 86.1 ± 0.6, 86.2 ± 0.2 and 93.0 ± 0.4 respectively. The AUROCs for cMLP are reported to be 86.6 ± 0.2, 96.6 ± 0.2 and 98.4 ± 0.1 respectively, while those for cLSTM are 81.3 ± 0.9, 93.4 ± 0.7 and 96.0 ± 0.1 respectively. NKDCD with ilg-lasso penalty gives AUROC of 94.1 ± 0.7, 98.5 ± 0.2 and 99.8 ± 0.1 while its counterpart with hlg-lasso is 94.1 ± 0.6, 98.5 ± 0.1 and 99.7 ± 0.1 respectively. When F is increased to 40, for $T = 250, 500, 1000$ the AUROCs for the VAR with ulg-lasso are 70.0 ± 0.0, 80.6 ± 0.0, and 84.7 ± 0.0, and with hlg-lasso are 70.4 ± 0.0, 81.1 ± 0.0, and 84.7 ± 0.0 respectively. The AUROCs for cMLP are reported to be 84.0 ± 0.5, 89.6 ± 0.2, and 95.5 ± 0.3, while those of cLSTM are 75.1 ± 0.9, 87.8 ± 0.4, and 94.4 ± 0.5 respectively. NKDCD with ilg-lasso penalty gives AUROC of 80.8 ± 0.1, 92.0 ± 1.1, and 98.7 ± 0.1 while its counterpart with hlg-lasso is 82.6 ± 0.3, 90.8 ± 0.7, and 96.7 ± 0.2 respectively.

4.3 fMRI Data with Nonlinear Dynamics



(a) Ground truth causal graph (b) Observed time series

Fig. 6. fMRI data

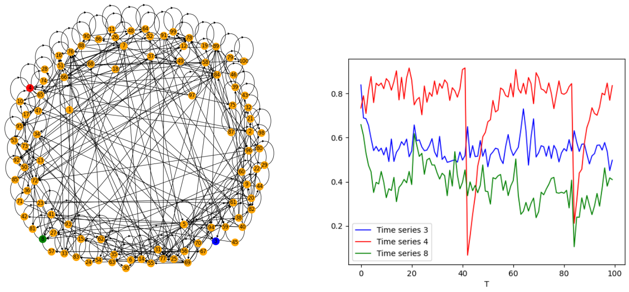
Another benchmark, functional magnetic resonance imaging (fMRI), contains realistic, simulated Blood-oxygen-level dependent (BOLD) datasets for 28 different underlying brain networks [41]. BOLD fMRI measures the neural activity of different regions of interest in the brain based on the changes in blood flow. Each region (i.e., a node in the brain network) has its associated time series of electrical excitation. All time series have an external input, white noise, and are fed through a nonlinear model [47]. All 28 datasets are also made available on GitHub. Brain network 20 has a dataset of size 2400×5 that corresponds to one of the 28 brain networks. Its ground truth causal graph and three selected time series are depicted in Figure 6, in which time series 2 and 3 exhibit a causal relationship, while neither is causally linked to time series 5.

TABLE 4
fMRI Results

Dataset	Brain network 20	
Metrics	AUROC	AUPR
NKDCD+Eq. 12	93.3 ± 0.0	93.0 ± 0.1
NKDCD+Eq. 11	95.3 ± 0.0	93.9 ± 0.1
VAR+Eq. 10	79.3 ± 3.9	82.5 ± 3.3
VAR+Eq. 11	79.3 ± 3.9	82.5 ± 3.3

The performance metrics are reported as the mean across five initializations with a 95% confidence interval. The results, summarized in Table 4, show that our formulation is promising and can be utilized in realistic datasets. The AUROC and AUPR for the standard VAR method using ulg- and hlg-lasso penalties are 79.3 ± 3.9 and 82.5 ± 3.3, respectively. Our method outperforms the VARs by having AUROC and AUPR of 93.3 ± 0.0 and 93.0 ± 0.1 for the ilg-lasso penalty and 95.3 ± 0.0 and 93.9 ± 0.1 respectively for the hlg- counterpart.

4.4 Dream3 Genetic Data with Nonlinear Dynamics



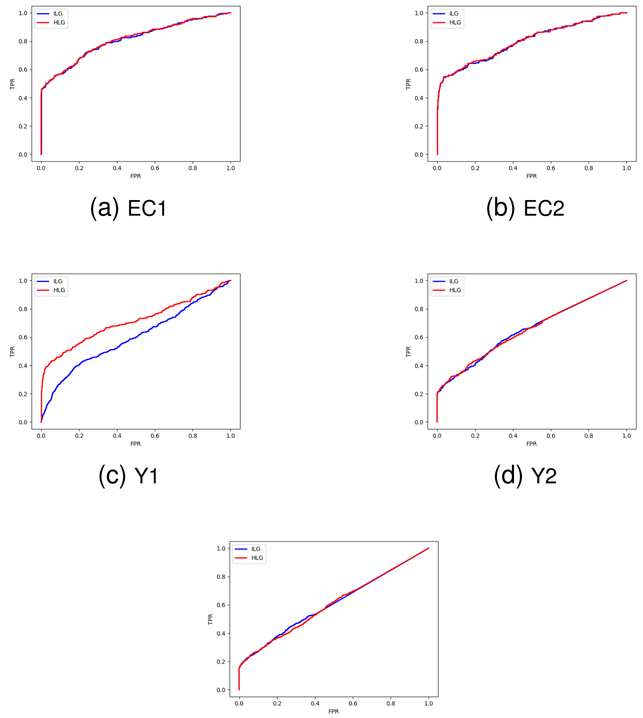
(a) Ground truth causal graph (b) Observed time series (selected)

Fig. 7. Dream3 genetic data for Yeast 1

TABLE 5
DREAM3 Genetic Data

	EC1		EC2		Y1		Y2		Y3	
Metric	AUROC	AUPR	AUROC	AUPR	AUROC	AUPR	AUROC	AUPR	AUROC	AUPR
NKDCD+Eq. 12	81.1	50.6	80.0	45.1	71.5	45.5	65.2	28.6	60.6	25.7
NKDCD+Eq. 11	81.2	50.6	80.2	45.5	71.6	45.6	65.1	29.2	60.4	25.6
cMLP	< 65	< 12	< 65	< 14	< 70	< 16	< 60	< 12	< 60	< 14
cLSTM	< 75	< 12	< 75	< 14	< 75	< 12	< 65	< 16	< 60	< 16
VAR+Eq. 10	51.8	2.6	50.9	2.6	50.2	2.9	51.3	5.3	52.4	7.6
VAR+Eq. 11	51.8	2.6	50.9	2.6	50.2	2.9	51.3	5.3	52.4	7.6

DREAM3 challenge dataset [42] provides a rigorous benchmark for comparing GC detection methods for gene regulatory networks [28], [48], [49]. The data are synthetically generated to mimic continuous gene expression and regulation dynamics, incorporating multiple unobserved hidden factors. The DREAM3 challenge comprises 5 distinct simulated datasets, each associated with different ground truth GC dependencies. Specifically, there are two datasets based on E.coli (EC 1-2) and three datasets based on Yeast (Y 1-3). In each dataset, there are 100 time series, with 46 replicates sampled at 21 time points, resulting in a times series data of size 100×966. This setup presents a significant challenge due to the limited amounts of data relative to the complexity of the networks and underlying dynamic interactions, as shown in Figure 7a, whereas 3 selected time series elements from a single replicate of the Y1 dataset are shown in Figure 7b. This snapshot provides insight into the intricacies of the gene expression dynamics encapsulated within the dataset, offering a stage for our GC estimation efforts. In this simulation, the ADAM [50] with parameters $\beta = (0.9, 0.999)$ and $\epsilon = 10^{-8}$, was used to initially update the learnable parameters in place of the ordinary gradient descent, followed by the proximal gradient descent was used to sparsify the lag matrices $W_l, l \in [1, L]$. We applied our results to all five data sets. Results for AUROC are visualized in Figure 8a and for AUPR in Figure 8b. The ROC plots for both the ilg- and hlg-lassos are shown in Figure 9.



(a) EC1

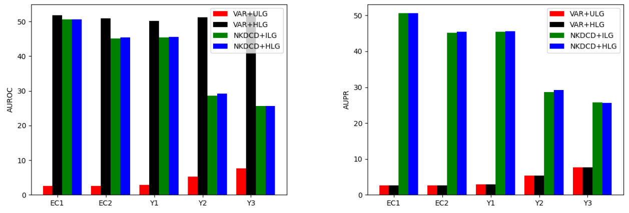
(b) EC2

(c) Y1

(d) Y2

(e) Y3

Fig. 9. ROC plots of DREAM3 dataset



(a) AUROC bar-plots

(b) AUPR bar-plots

Fig. 8. Accuracy plots of DREAM3 dataset

We compared our results based on the bar plots in the recent work reported in [28] in Table 5. For the EC1 dataset, the AUROC values for the standrad VAR with ulg-lasso and with hlg-lasso, cMLP, cLSTM, NKDCD with ilg-lasso, and with hlg-lasso respectively are $51.8, 51.8$,65, 75, 81.1, 81.2 and their corresponding AUPRs are 2.6, 2.6, 12, 12, 50.6, 50.6, respectively. Following the same pattern for the EC2 dataset, the AUROCs are 50.9, 50.9,65, 75, 80.0, 80.2 and their corresponding AUPRs are 2.6, 2.6, 14, 14, 45.1, 41.5, respectively. The performance of the Y1 dataset for AUROCs are 50.2, 50.2,70, 75, 71.5, 71.6 and their corresponding AUPRs are 2.9, 2.9, 16, 12, 45.5, 45.6, respectively. The Y2 dataset yields AUROCs are as, 51.3, 51.3,60, 65, 65.2, 65.1 and their corresponding AUPRs as, 5.3, 5.3, 12, 16, 28.6, 29.2, respectively. The Y3 Dataset gives AUROCs as, 52.4, 52.4,60, 60, 60.6, 60.4 and their corresponding AUPRs as, 7.6, 7.6, < 14, < 16, 25.7, 25.6, respectively. Results suggest that the dynamics for Y 1-3 may be more complex than that for EC 1-2. In all cases, our results outperform the existing linear and nonlinear methods.

5 Conclusion

We have introduced the NeuroKoopman Dynamic Causal Discovery (NKDCD) framework, which leverages a Koopman-inspired regularized deep neural network autoencoder architecture for the inference of nonlinear Granger causality and underlying dynamics in observed multivariate time series data. This framework enables the learning of underlying dynamics and their interdependencies through sparse nonlinear regressive models. By employing a data-driven NN-based basis function for lifting the NAR dependency for a linear Koopman embedding and next for learning a linear vector autoregressive model in the lifted domain, we infer Granger causality and the underlying dynamics. We employ structured group-level sparsity-inducing penalties that preserve the structure of interdependencies of the original variables, ensuring reliable model learning. Validation through simulations on various application datasets shows that our NKDCD outperforms the state-of-the-art results from recent literature.

Bibliography & References Cited

[1] R. Guo, L. Cheng, J. Li, P. R. Hahn, and H. Liu, "A survey of learning causality with data: Problems and methods," *ACM Computing Surveys (CSUR)*, vol. 53, no. 4, pp. 1–37, 2020.

[2] C. W. Granger, "Investigating causal relations by econometric models and cross-spectral methods," *Econometrica: Journal of the Econometric Society*, pp. 424–438, 1969.

[3] H. Lütkepohl, *New Introduction to Multiple Time Series Analysis*. Springer Science & Business Media, 2005.

[4] Y. Hong, Y. Liu, and S. Wang, "Granger causality in risk and detection of extreme risk spillover between financial markets," *Journal of Econometrics*, vol. 150, no. 2, pp. 271–287, 2009.

[5] A. T. Reid, D. B. Headley, R. D. Mill, R. Sanchez-Romero, L. Q. Uddin, D. Marinazzo, D. J. Lurie, P. A. Valdés-Sosa, S. J. Hanson, B. B. Biswal *et al.*, "Advancing functional connectivity research from association to causation," *Nature neuroscience*, vol. 22, no. 11, pp. 1751–1760, 2019.

[6] T. J. Mosedale, D. B. Stephenson, M. Collins, and T. C. Mills, "Granger causality of coupled climate processes: Ocean feedback on the north atlantic oscillation," *Journal of climate*, vol. 19, no. 7, pp. 1182–1194, 2006.

[7] S. Z. Chiou-Wei, C.-F. Chen, and Z. Zhu, "Economic growth and energy consumption revisited—evidence from linear and nonlinear granger causality," *Energy economics*, vol. 30, no. 6, pp. 3063–3076, 2008.

[8] P. A. Stokes and P. L. Purdon, "A study of problems encountered in Granger causality analysis from a neuroscience perspective," *Proceedings of the National Academy of Sciences*, vol. 114, no. 34, pp. E7063–E7072, 2017.

[9] A. Sheikhattar, S. Miran, J. Liu, J. B. Fritz, S. A. Shamma, P. O. Kanold, and B. Babadi, "Extracting neuronal functional network dynamics via adaptive Granger causality analysis," *Proceedings of the National Academy of Sciences*, vol. 115, no. 17, pp. E3869–E3878, 2018.

[10] A. Fujita, P. Severino, J. R. Sato, and S. Miyano, "Granger causality in systems biology: modeling gene networks in time series microarray data using vector autoregressive models," in *Brazilian Symposium on Bioinformatics*. Springer, 2010, pp. 13–24.

[11] A. C. Lozano, N. Abe, Y. Liu, and S. Rosset, "Grouped graphical granger modeling methods for temporal causal modeling," in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2009, pp. 577–586.

[12] C. A. Sims, "Money, income, and causality," *The American economic review*, vol. 62, no. 4, pp. 540–552, 1972.

[13] G. Chamberlain, "The general equivalence of granger and sims causality," *Econometrica: Journal of the Econometric Society*, pp. 569–581, 1982.

[14] J. B. Cromwell, *Multivariate tests for time series models*. Sage, 1994, no. 100.

[15] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[16] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68, no. 1, pp. 49–67, 2006.

[17] W. B. Nicholson, J. Bien, and D. S. Matteson, "Hierarchical vector autoregression," *arXiv preprint arXiv:1412.5250*, 2014.

[18] A. Shojaie and G. Michailidis, "Discovering graphical Granger causality using the truncating lasso penalty," *Bioinformatics*, vol. 26, no. 18, pp. i517–i523, 2010.

[19] T. Teräsvirta, D. Tjøstheim, C. W. J. Granger *et al.*, *Modelling Nonlinear Economic Time Series*. Oxford University Press Oxford, 2010.

[20] B. Lusch, P. D. Maia, and J. N. Kutz, "Inferring connectivity in networked dynamical systems: Challenges using Granger causality," *Physical Review E*, vol. 94, no. 3, p. 032220, 2016.

[21] R. Vicente, M. Wibral, M. Lindner, and G. Pipa, "Transfer entropy—a model-free measure of effective connectivity for the neurosciences," *Journal of Computational Neuroscience*, vol. 30, no. 1, pp. 45–67, 2011.

[22] P.-O. Amblard and O. J. Michel, "On directed information theory and Granger causality graphs," *Journal of Computational Neuroscience*, vol. 30, no. 1, pp. 7–16, 2011.

[23] J. Runge, J. Heitzig, V. Petoukhov, and J. Kurths, "Escaping the curse of dimensionality in estimating multivariate transfer entropy," *Physical Review Letters*, vol. 108, no. 25, p. 258701, 2012.

[24] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Multistep neural networks for data-driven discovery of nonlinear dynamical systems," *arXiv preprint arXiv:1801.01236*, 2018.

[25] Ö. Kişi, "River flow modeling using artificial neural networks," *Journal of Hydrologic Engineering*, vol. 9, no. 1, pp. 60–63, 2004.

[26] A. Graves, "Supervised sequence labelling," in *Supervised Sequence Labelling with Recurrent Neural Networks*. Springer, 2012, pp. 5–13.

[27] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Graph convolutional recurrent neural network: Data-driven traffic forecasting," *arXiv preprint arXiv:1707.01926*, 2017.

[28] A. Tank, I. Covert, N. Foti, A. Shojaie, and E. B. Fox, "Neural granger causality," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 8, pp. 4267–4279, 2021.

[29] B. O. Koopman, "Hamiltonian systems and transformation in hilbert space," *Proceedings of the National Academy of Sciences*, vol. 17, no. 5, pp. 315–318, 1931.

[30] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, "A data–driven approximation of the koopman operator: Extending dynamic mode decomposition," *Journal of Nonlinear Science*, vol. 25, pp. 1307–1346, 2015.

[31] P. J. Schmid, "Dynamic mode decomposition and its variants," *Annual Review of Fluid Mechanics*, vol. 54, pp. 225–254, 2022.

[32] M. Korda and I. Mezić, "Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control," *Automatica*, vol. 93, pp. 149–160, 2018.

[33] S. Sinha and U. Vaidya, "On data-driven computation of information transfer for causal inference in discrete-time dynamical systems," *Journal of Nonlinear Science*, vol. 30, no. 4, pp. 1651–1676, 2020.

[34] R. Gunjal, S. S. Nayyer, S. Wagh, A. Stankovic, and N. Singh, "Granger causality for prediction in dynamic mode decomposition: Application to power systems," *Electric Power Systems Research*, vol. 225, p. 109865, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378779623007538

[35] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[36] C. Wehmeyer and F. Noé, "Time-lagged autoencoders: Deep learning of slow collective variables for molecular kinetics," *The Journal of chemical physics*, vol. 148, no. 24, 2018.

[37] B. Lusch, J. N. Kutz, and S. L. Brunton, "Deep learning for universal linear embeddings of nonlinear dynamics," *Nature communications*, vol. 9, no. 1, p. 4950, 2018.

[38] R. R. Hossain, R. Adesunkanmi, and R. Kumar, "Data-driven linear koopman embedding for networked systems: Model-predictive grid control," *IEEE Systems Journal*, 2023.

[39] M. Nauta, D. Bucur, and C. Seifert, "Causal discovery with attention-based convolutional neural networks," *Machine Learning and Knowledge Extraction*, vol. 1, no. 1, pp. 312–340, 2019.

[40] E. N. Lorenz, "Predictability: A problem partly solved," in *Proc. Seminar on predictability*, vol. 1, no. 1. Reading, 1996.

[41] S. M. Smith, K. L. Miller, G. Salimi-Khorshidi, M. Webster, C. F. Beckmann, T. E. Nichols, J. D. Ramsey, and M. W. Woolrich, "Network modelling methods for fmri," *Neuroimage*, vol. 54, no. 2, pp. 875–891, 2011.

[42] R. J. Prill, D. Marbach, J. Saez-Rodriguez, P. K. Sorger, L. G. Alexopoulos, X. Xue, N. D. Clarke, G. Altan-Bonnet, and G. Stolovitzky, "Towards a rigorous assessment of systems biology models: the dream3 challenges," *PloS One*, vol. 5, no. 2, p. e9202, 2010.

[43] A. C. Lozano, N. Abe, Y. Liu, and S. Rosset, “Grouped graphical granger modeling for gene expression regulatory networks discovery,” *Bioinformatics*, vol. 25, no. 12, pp. i110–i118, 2009.

[44] A. Tank, I. Covert, N. Foti, A. Shojaie, and E. Fox, “Neural granger causality,” *arXiv preprint arXiv:1802.05842*, 2018.

[45] N. Parikh, S. Boyd *et al.*, “Proximal algorithms,” *Foundations and Trends in Optimization*, vol. 1, no. 3, pp. 127–239, 2014.

[46] E. F. Fama and K. R. French, “The cross-section of expected stock returns,” *the Journal of Finance*, vol. 47, no. 2, pp. 427–465, 1992.

[47] R. B. Buxton, E. C. Wong, and L. R. Frank, “Dynamics of blood flow and oxygenation changes during brain activation: the balloon model,” *Magnetic resonance in medicine*, vol. 39, no. 6, pp. 855–864, 1998.

[48] N. Lim, F. d’Alché Buc, C. Auliac, and G. Michailidis, “Operator-valued kernel-based vector autoregressive models for network inference,” *Machine learning*, vol. 99, pp. 489–513, 2015.

[49] S. Lèbre, “Inferring dynamic genetic networks with low order independencies,” *Statistical applications in genetics and molecular biology*, vol. 8, no. 1, 2009.

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

Rahmat Adesunkanmi is a Ph.D. student at the Department of Electrical and Computer Engineering, Iowa State University, Ames, USA. She received her B.S. degree from the University of Ibadan, Nigeria, in 2015 and subsequently her master’s degree from Iowa State University. Her research interests include noise-robust Machine learning techniques and Time-series analysis. She is an active member of the Graduate Society for Women Engineers.

Balaji Sesha Srikanth Pokuri received the B.Tech degree in electrical engineering from Indian Institute of Technology, Roorkee, India, in 2017. He is currently working toward the Ph.D. degree in electrical engineering with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA, USA.

He worked as a Quantitative Analyst in eFX Trading business of HSBC Global Markets, from 2017 to 2021. His current research includes data-driven modeling, control, optimization, and artificial-intelligence-based approaches for precision agriculture.

Ratnesh Kumar is a Palmer Professor at the Department of Electrical and Computer Engineering, Iowa State University, where he directs the **ESSeNCE** (**E**mbedded **S**oftware, **Se**nsors, **N**etworks, **C**yberphysical, and **E**nergy) Lab. Previously, he held a faculty position at the University of Kentucky and various visiting positions with the University of Maryland (College Park), the Applied Research Laboratory at the Pennsylvania State University (State College), NASA Ames, the Idaho National Laboratory, the United Technologies Research Center, and the Air Force Research Laboratory. He received a B. Tech. degree in Electrical Engineering from IIT Kanpur, India, in 1987 and the M.S. and Ph.D. degrees in Electrical and Computer Engineering from the University of Texas at Austin in 1989 and 1991 respectively. Ratnesh is a Fellow of IEEE, also a Fellow of AAAS, and was a Distinguished Lecturer of the IEEE Control Systems Society. He is a recipient of *D. R. Boylan Eminent Faculty Award for Research* and *Award for Outstanding Achievement in Research* from Iowa State University, and also the *Distinguished Alumni Award* from IIT Kanpur. Ratnesh received Gold Medals for the Best EE Undergrad, the Best All Rounder, and the Best EE Project from IIT Kanpur, and the Best Dissertation Award from UT Austin, the Best Paper Award from the IEEE Transactions on Automation Science and Engineering, and has been Keynote Speaker and paper award recipient from multiple conferences. He is or has been an editor of several journals (including of IEEE, SIAM, ACM, Springer, IET, MDPI).