Homagni Saha
Department of Mechanical Engineering,
Iowa State University,
Ames, IA 50011
e-mail: hsaha@iastate.edu

Chao Liu
Energy and Power Engineering,
Tsinghua University,
Beijing 100084, China

Zhanhong Jiang
Johnson Controls,
Milwaukee, WI 53202

Soumik Sarkar[1]
Mem. ASME
Department of Mechanical Engineering,
Iowa State University,
Ames, IA 50011
e-mail: soumiks@iastate.edu

Data-Driven Performance Monitoring of Dynamical Systems Using Granger Causal Graphical Models

Data-driven analysis and monitoring of complex dynamical systems have been gaining popularity due to various reasons like ubiquitous sensing and advanced computation capabilities. A key rationale is that such systems inherently have high dimensionality and feature complex subsystem interactions due to which majority of the first-principle based methods become insufficient. We explore the family of a recently proposed probabilistic graphical modeling technique, called spatiotemporal pattern network (STPN) in order to capture the Granger causal relationships among observations in a dynamical system. We also show that this technique can be used for anomaly detection and root-cause analysis for real-life dynamical systems. In this context, we introduce the notion of Granger-STPN (G-STPN) inspired by the notion of Granger causality and introduce a new nonparametric technique to detect causality among dynamical systems observations. We experimentally validate our framework for detecting anomalies and analyzing root causes in a robotic arm platform and obtain superior results compared to when other causality metrics were used in previous frameworks. [DOI: 10.1115/1.4046673]

1 Introduction

A wide variety of human-engineered applications leverage large scale cyber-physical systems (CPSs), such as integrated buildings [1], transportation networks [2], robotic networks [3], smart home internet of things (IoT) [4], and wind farms [5]. For decision and control, it is crucial to understand the interactions among different parts or subsystems of such large-scale systems. Although it is possible to model the interactions in detail, which most physics-based models do using first principles, it becomes highly complex with increasing number of subsystems. Therefore, data-driven methods have been receiving considerable attention from the industry and academia being more scalable and accurate. However, data-driven modeling of these spatiotemporal (causal) interactions are not trivial and is a crucial step for performance monitoring and diagnostics as well as developing advanced control for large-scale CPSs. Information theoretic techniques can help in this regard, e.g., Granger causality (a causality metric that works on the hypothesis that a process X is causal to Y if predictions about Y made using joint history of X and Y is better than those made using only history of Y) can provide relevant insights when considering the effectiveness of control mechanisms [6]. Although research in finance [7], neuroscience [8], and social sciences [9] has focused on identification of such causal interactions, for large scale CPSs, the applications have not been explored sufficiently.

A recently proposed probabilistic graphical modeling technique—spatiotemporal pattern network (STPN) has been used for modeling the multivariate time series observations from distributed CPSs, with reasonable success. The proposed framework was formulated using symbolic dynamic filtering (SDF) [10], with applications in diagnostics and root-cause analysis of physical faults and cyber anomalies in CPSs [11–14] residential energy disaggregation [1], building occupancy detection [15], and wind energy prediction [5]. While the method has been shown to be effective in practice, so far there has been no rigorous analysis on whether it is able to capture causality among observations from the subsystems. This paper explores the capability of STPN in capturing Granger causality among observations in a dynamical system. In this context, we introduce two variants of STPN, namely, transfer-STPN (T-STPN), and Granger-STPN (G-STPN). T-STPN leverages the concept of transfer entropy computed between two symbolic time series to detect causality. On the other hand, using the G-STPN framework, we propose a new nonlinear causality detection metric which extends naturally from the STPN framework. In T-STPN, transfer entropy is used to quantify the dependency between two observations as opposed to mutual information that was originally used in the STPN framework [1].[2]

The key difference across the three formulations, STPN, T-STPN, and G-STPN is the way in which the underlying state spaces of the two time series under consideration are formulated. In both G-STPN and T-STPN, (joint) product state space is considered as opposed to individual state spaces that were used in STPN for improved performance. The remainder of the paper is organized as follows: Sec. 2 provides preliminaries on two main approaches for causality detection, Sec. 3 introduces STPN, T-STPN, and G-STPN frameworks—our proposed improvement upon two existing popular data driven frameworks for anomaly detection and performance monitoring. After that, Sec. 4 briefly provided the series of steps involved for anomaly detection and root cause analysis using our framework. In Sec. 5, we perform an empirical study and compare our newly proposed causality detection metric with transfer entropy. In Sec. 6, we provide details of our experimental setup on a real seven degrees-of-freedom (7DOF) robotic manipulator, followed by performance and results in Sec. 7 (overview of our results provided in Fig. 1).

Contributions: In summary, the overall contributions of this paper are: (i) we propose a generalized G-STPN graphical modeling framework (detailed flowchart provided in Fig. 2) and define a

[1]Corresponding author.

Contributed by the Dynamic Systems Division of ASME for publication in the Journal of Dynamic Systems, Measurement, and Control. Manuscript received November 18, 2019; final manuscript received March 3, 2020; published online April 6, 2020. Assoc. Editor: Alessandro Rizzo.

[2]Note that in a preliminary study [12], we used the term G-STPN for the STPN variant with transfer entropy. However, in this paper we refer to this variant as T-STPN as we call our new framework as G-STPN where we propose a new metric comparable to Granger causality.

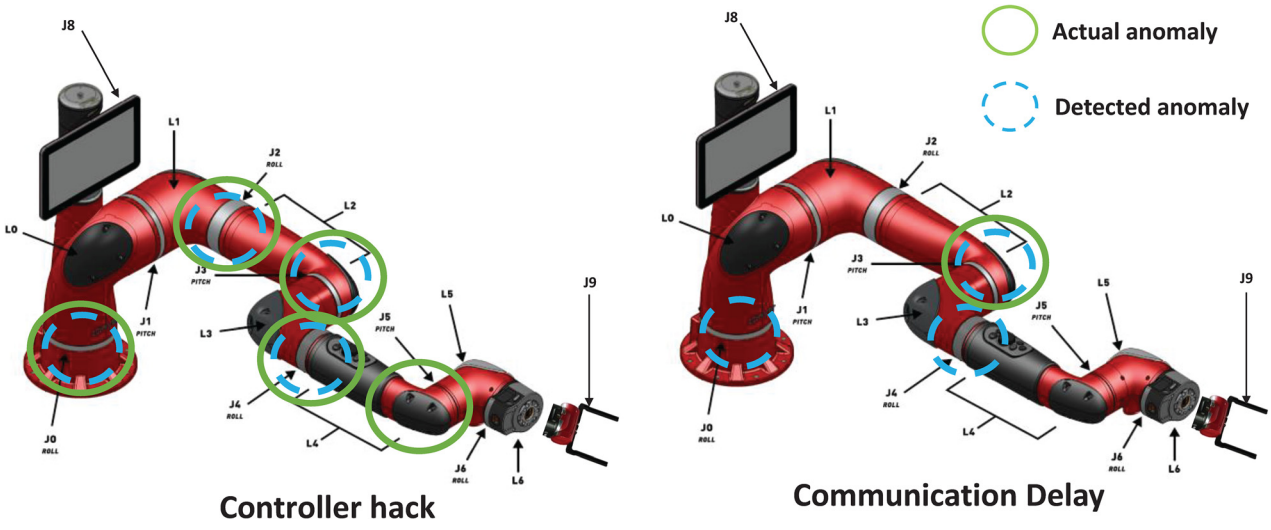

Fig. 1 We develop an anomaly detection and root cause analysis framework for CPSs. This figure shows the results of our algorithm on correctly identifying injected cyber-physical attacks in a complex 7DOF manipulator in two types of crafted attacks-controller hack and communication delay (explained in the experiments section).

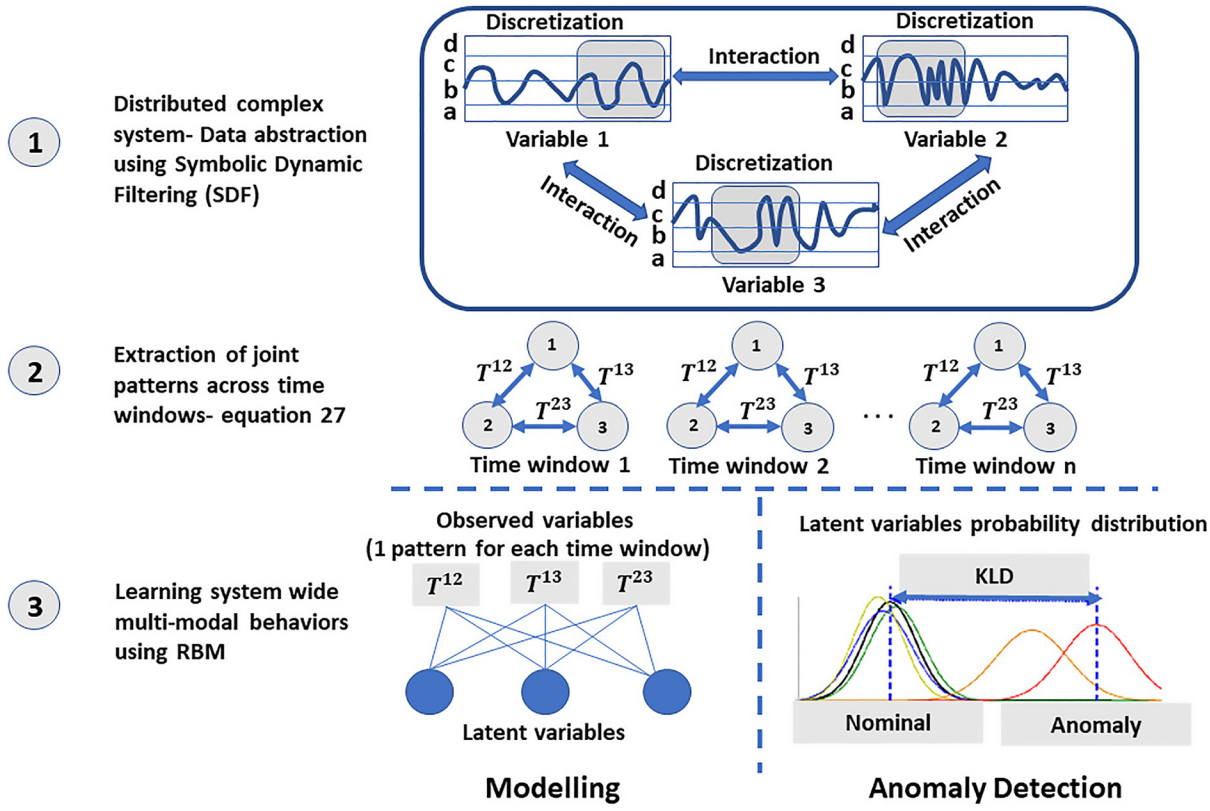

Fig. 2 Detailed flowchart explaining the G-STPN framework

new nonlinear relationship metric comparable to Granger causality; (ii) we experimentally validate the G-STPN framework in a performance monitoring problem of an industrial manipulator (robotic arm), we present detailed anomaly detection and root cause analysis performance of G-STPN along with comparison with previous approaches, STPN and T-STPN.

2 Multivariate Feature Extraction and Association

Multivariate feature extraction is an important step for information fusion uniting research in several directions such as anomaly detection, event classification [16] and even visualizing multi-agent or multimode communication in robotics [17,18]. In this section, we will briefly describe the techniques that we use for converting real-valued multivariate time series data into useful features and how those features are associated with each other in our causal framework—G-STPN. First, we use SDF based encoding to convert real valued time series into sequence of symbols. Details are described in supplementary section available in the Supplemental Materials on the ASME Digital Collection. Then, we extract states of the system by using time delay embedding. Thereafter, we use joint state information from a pair of variables

to predict causal influence on a target, variable giving us an estimate of association between the particular states of the two variables. This forms the backbone of our G-STPN framework. A short schematic is provided in Fig. 3—on the left, the embedding process is visualized, and on the right the association between states for both STPN and G-STPN is visualized. Following Fig. 3, let us consider two time series variables I and J. Let the observation at the $(t+1)$th instant of sequence I be i_{t+1}, which depends on its previous state, $\vec{i}_t^{\,k} := \{i_{t-k}, \ldots, i_{t-1}, i_t\}$, and accordingly for variable J. Then we can use the following equation for finding transfer entropy from I to J, which is also a data driven nonlinear measure of Granger causality

$$\tau_{I\to J} = \sum p\left(j_{t+1}, \vec{j}_t^{\,k}, \vec{i}_t^{\,k}\right) \log\left(\frac{p\left(j_{t+1}|\vec{j}_t^{\,k}, \vec{i}_t^{\,k}\right)}{\log\left(p\left(j_{t+1}|\vec{j}_t^{\,k}\right)\right)}\right) \tag{1}$$

Detailed derivation of Eq. (1) is provided in supplementary section available in the Supplemental Materials on the ASME Digital Collection. After having associated all pairs of variables in the system through derived transfer entropy, we proceed to defining our framework in Sec. 3.

3 Granger-Spatiotemporal Pattern Network Framework

In this section we generalize STPN framework as G-STPN which includes nonlinear metrics for causality detection which is akin to Granger causality.

3.1 Granger-Spatiotemporal Pattern Network Framework. An STPN is intended to capture the strengths of "causality" between different nodes of the graph which correspond to the variables of a multivariate time series under observation. While details of STPN can be found in our previous work [1], we provide the definition of STPN in supplementary section, available in the Supplemental Materials on the ASME Digital Collection, for completeness. Here we propose a new variant of STPN, called G-STPN.

DEFINITION 3.1. *A Granger-STPN is a 5-tuple* $W = (Q^I, Q^J, \Sigma^J, \tau^{IJ}, T^{IJ})$*: (I, J denote nodes of the Granger-STPN which are basically two different variables of a multivariate time series where causality is investigated)*

(1) $Q^I = \{q_1, q_2, \ldots, q_{|Q_I|}\}$ *is the state set corresponding to all k-lag embeddings of symbol sequences* S^I*;*
(2) $Q^J = \{q_1, q_2, \ldots, q_{|Q_J|}\}$ *is the state set corresponding to all k-lag embeddings of symbol sequences* S^J*;*
(3) $\Sigma^J = \{\sigma_0, \ldots, \sigma_{|\Sigma^J|-1}\}$ *is the alphabet set of symbol sequence* S^J*;*
(4) τ^{IJ} *is the joint state-symbol generation matrix of size* $|Q^I| \times |Q^J| \times |\Sigma^J|$*, the ijth element of* τ^{IJ} *denotes the probability of finding the symbol* σ_j *in the symbol string* S^J *while making a transition from the state* $q_i \in Q^I$ *and (jointly) state* $q_j \in Q^J$ *such a pattern is called joint pattern between nodes of I and J; and*
(5) T^{IJ} *denotes a metric that can represent causality in a certain time window between I and J (degree of influence of variation of J on I), denoted* $I \to J$ *which is a function of* τ^{IJ}*.*

As STPN only uses individual state spaces to describe atomic patterns and relational patterns, we cannot compare them without rigorous normalization processes. Therefore, we consider variants of the STPN framework, called G-STPN and T-STPN that consider the joint/product state space. In this case, the relational patterns in STPN get replaced by the joint patterns as shown in Fig. 3. While we use transfer entropy as $T^{IJ} = \tau^{IJ}$ in T-STPN, a new Granger causality based metric is defined for G-STPN which we will now denote as T^{IJ} in Secs. 3.2, 3.3, and 4–7. While modeling the joint state symbol generation matrix, we would require a mechanism to prevent the dimensions of the matrix τ^{IJ} from increasing dramatically with increasing number states in Q^I, Q^J and alphabets in Σ^J. In order to achieve this, we use the state merging algorithm which is explained in detail in Ref. [1] as well as in the supplementary section available in the Supplemental Materials on the ASME Digital Collection. We need to find a joint symbol state generation matrix for each variable pair $(x, y) \in (I, J)$, where (I, J) is the set of all pairs of variables in the multivariate time series. Therefore, we model the matrix τ^{xy} using the Dirichlet distribution following previous works [19], and find out an expression for its prior joint density conditioned on a realization (x_n, y_n) of n data-points from two variables of the time series X and Y. We have included detailed steps for deriving this joint probability distribution in the

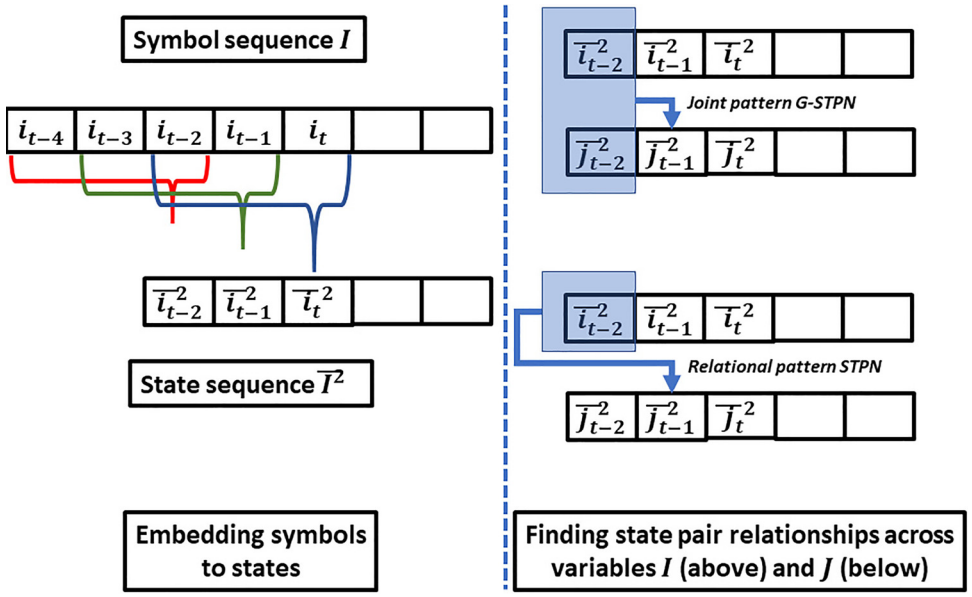

Fig. 3 Left—demonstrating the process of embedding a symbol sequence *I* into a state sequence $\overline{I}^2$, with each *state* having an embedding dimension $k = 2$. Right—using obtained state sequences in the STPN and G-STPN frameworks.

supplementary section available in the Supplemental Materials on the ASME Digital Collection.

3.2 Inferring About Test Observations. Granger-STPN is a pattern based framework that can capture multiple operating modes of a system. A graphical model learned at a time instant may not be same as the graphical model learned at another instant, especially when the system moves on to a different mode of operation. However, this change can be recorded as patterns in the graph connectivity. The role of importance metric T defined in earlier sections is to capture these changes in flow of information from one node of the graphical model to another. However, there is no direct way to infer this change. One way to go about it is to evaluate the likelihood of a subsequence of data based on past observed data. We first model the nominal distribution of the data through τ^{xy}. During online inference, we calculate the likelihood of an observed small time window of the data based on our prior model. We denote variables collected during inference phase with a tilde symbol in the superscript. For an observed subsequence of symbols in process Y, we denote the joint symbols of X and Y collected in training stage as $\mathcal{S}_\tau$, symbols of Y collected in training stage as $\mathcal{S}_\pi$ and those in inference stage as $\tilde{\mathcal{S}}$. We also use t and k to denote the time point of observed variables in the inference and training sequences, respectively. We call X as the source time series and Y as the target time series. We are interested in determining the following two probabilities:

(1) Probability that a probabilistic finite state automaton with transition matrix τ^{xy} and joint state set of $|Q^X| \times |Q^Y|$ generated the subsequence $\tilde{\mathcal{S}}$. We call the model as full (joint) model, and denote the probability as Λ^{xy}.
(2) Probability that a probabilistic finite state automaton with transition matrix Π^y and a state set of $|Q^Y|$ generated the subsequence $\tilde{\mathcal{S}}$. We call the model as reduced (self) model, and denote the probability as λ^y. Note that in this case $\tilde{\mathcal{S}}$ does not depend on X.

We therefore define Λ and λ as

$$\Lambda^{xy} = \Pr(\tilde{Q}_t^x, \tilde{Q}_t^y, \tilde{S}_{t+1}^y | Q_k^x, Q_k^y, S_{k+1}^y) \equiv \Pr(\tilde{\mathcal{S}}|\mathcal{S}_\tau) \tag{2}$$

$$\lambda^y = \Pr(\tilde{Q}_t^y, \tilde{S}_{t+1}^y | Q_k^y, S_{k+1}^y) \equiv \Pr(\tilde{\mathcal{S}}|\mathcal{S}_\pi) \tag{3}$$

We obtain a set of patterns Λ^{xy} and λ^y, for all x and y forming pairs of time series in the multivariate time series of the system. Let $\tilde{N}_{mn}^{Yy}$ denote the number of times in the short subsequence that the symbol σ_n^y was observed while there was a state $q_m \in |Q^Y|$. Then $\tilde{N}_m^{Yy}$ denotes the number of times the state $q_m \in |Q^Y|$ was observed in the short subsequence. Thus, the probability that the "self-model" generated $\tilde{\mathcal{S}}$ is given by the product of independent multinomial distributions

$$\Pr(\tilde{\mathcal{S}}|\Pi^y) = \prod_{m=1}^{|Q^y|} \left(N_m^{\tilde{Y}y}\right)! \prod_{n=1}^{|\Sigma^y|} \frac{(\Pi_{mn}^y)^{\tilde{N}_{mn}^{Yy}}}{\tilde{N}_{mn}^{Yy}!} \tag{4}$$

The results from the testing are now conditioned on the training data. Given the symbol string $\mathcal{S}_\pi$ in the training phase the probability of observing the symbol string $\tilde{\mathcal{S}}$ is given by

$$\Pr(\tilde{\mathcal{S}}|\mathcal{S}_\pi) = \lambda^y = \prod_{m=1}^{|Q^y|} \frac{\left(\tilde{N}_m^{Yy}\right)!(N_m^{Yy} + |\Sigma^y| - 1)!}{\left(\tilde{N}_m^{Yy} + N_m^{Yy} + |\Sigma^y| - 1\right)!} \times \prod_{n=1}^{|\Sigma^y|} \frac{\left(\tilde{N}_{mn}^{Yy} + N_{mn}^{Yy}\right)!}{\left(\tilde{N}_{mn}^{Yy}\right)!(N_{mn}^{Yy})!} \tag{5}$$

Details of deriving the equation above is provided in the supplementary section available in the Supplemental Materials on the ASME Digital Collection. The probability of the joint state-symbol subsequence is also a product of independent multinomial distributions given that the exact joint state symbol generation matrix is known

$$\Pr(\tilde{S}|\tau^{xy}) = \prod_{m=1}^{|Q^x|\times|Q^y|} \left(\tilde{N}_m^{xy}\right)! \prod_{n=1}^{|\Sigma^y|} \frac{(\tau_{mn}^{xy})^{\tilde{N}_{mn}^{xy}}}{\tilde{N}_{mn}^{xy}!} \tag{6}$$

where the definition of $\tilde{N}_{mn}^{xy}$ is similar to N_{mn}^{xy} in the context of short subsequence. With the similar derivation as above, which can be also seen in Ref. [19], the metric $\Lambda^{xy}(\tilde{Q}^x, \tilde{Q}^y, \tilde{S}^y)$ can be obtained as follows:

$$\Pr(\tilde{\mathcal{S}}|\mathcal{S}_\tau) = \Lambda^{xy} = \prod_{m=1}^{|Q^x|\times|Q^y|} \frac{\left(\tilde{N}_m^{xy}\right)!(N_m^{xy} + |\Sigma^y| - 1)!}{\left(\tilde{N}_m^{xy} + N_m^{xy} + |\Sigma^y| - 1\right)!} \times \prod_{n=1}^{|\Sigma^y|} \frac{\left(\tilde{N}_{mn}^{xy} + N_{mn}^{xy}\right)!}{\left(\tilde{N}_{mn}^{xy}\right)!(N_{mn}^{xy})!} \tag{7}$$

3.3 Granger Causal Online Inference Metric. As a pattern based algorithm, we use a metric to capture how important is the interaction between two time series at a given instant (for a dynamical system having multiple time series). When inferring about the nature of the observed data, we consider short time sequences which are collected as the system operates. For each pair of variables X and Y in the n-point time window of (x_n, y_n), we calculate the importance metric. The importance metric T^{xy} should have two desirable properties:

(1) It should reflect the degree to which the full model (the model learned by using joint state space consideration of the target time series and the source time series which is suspected to have influence on the target time series, or joint patterns in short) yields a better prediction of the target variable than the reduced model (the model learned by using the state space of the target time series only, or atomic patterns in short), as inferred from the given time window.
(2) It should easily reflect the importance of the current observed test (joint) pattern with respect to the learned nominal pattern τ^{xy} in the modeling phase.

As a framework that detects anomalies based on dynamics of changes in the influence that one time series exerts on another (in a multivariate time series setting), we believe that property 1 is more important than property 2. In this regard, transfer entropy has been used successfully in several applications [7–9] to estimate this desirable property 1. However, a tricky issue in evaluating this empirical metric is to obtain accurate estimates of conditional and joint probabilities to compute conditional entropy values. To preserve property 1, it is very convenient to use transfer entropy as a direct estimator. However, here we will take a detour and explore a new concept to detect (possibly) a wide range of causality in a nonparametric manner, thus establishing a metric alongside the existing transfer entropy.

Based on their respective formulation, it is evident that Λ should be always much smaller than λ. This is because one involves three dimensions (states of X, states of Y and future symbols of Y), whereas the other only involves two dimensions (states of Y and future symbols of Y). Therefore, the two probabilities cannot be compared unless we introduce a baseline: "if the process corresponding to X does not cause the process corresponding to Y then for the baseline, the joint probability Λ can be written as individual products of self-probability λ over all possible states of X." This gives rise to the following baseline:

$$\log(\hat{\Lambda}^{xy}) \triangleq |Q^x| \log(\lambda^y) \text{ when } X \text{ causes } Y \tag{8}$$

$$\log(\hat{\Lambda}^{xy}) \approx \log(\Lambda^{xy}) \text{ when } X \text{ does not cause } Y \tag{9}$$

However, if causality does hold, then our assumption does not apply and we quantify our causality detection metric as

$$T^{xy} = \log(\hat{\Lambda}^{xy}) - \log(\Lambda^{xy}) \tag{10}$$

In order to test our newly proposed metric, we have performed empirical analysis of our metric on several example time series pairs modeled with several different types of causal structures within them. We found out that our metric is at least as sensitive as the data driven transfer entropy metric. Detailed results and explanations are provided in the supplementary section available in the Supplemental Materials on the ASME Digital Collection.

4 Spatiotemporal Pattern Network for Anomaly Detection and Root-Cause Analysis

After constructing G-STPN using multivariate time series data, a learning framework using a combination of G-STPN and restricted Boltzmann machine (RBM) as introduced in Refs. [11], [12], and [20] can be leveraged for anomaly detection and root-cause analysis. In brief, RBMs are used to effectively capture a probability distribution for sequences of observed spatiotemporal patterns. The observed patterns during "nominal" operation of a system are assigned high probability of occurrence (highly likely to happen) which is inversely proportional to a metric associated with trained RBMs called "free-energy." If free energy for an input sequence pattern to an RBM is low, the probability of occurrence of that sequence is high. Conversely, if the free energy is high, then the probability of occurrence is low. When an anomalous operation is observed, it causes an high free energy output. The sequential state switching (S^3) algorithm [11] takes advantage of this fact and manipulates (flip from 0 to 1 and vice versa) each bit of the input sequence one by one to revert the free energy output to a low value. The corresponding bits which when flipped reverts the RBM output to low free energy are the potential "root-causes" for the anomaly. Interested readers are suggested to read [11,20] for further details. The overall process is briefly explained below:

(1) Perform discretization (e.g., maximum entropy partitioning [20]) of given multivariate time series into a N number of bins and symbolize the time series as decided by the strategy described in Sec. 2;
(2) For each variable a, find the state Π^q_{ij} which denotes the probability of transition from state i in state sequence a to symbol j in symbol sequence a;
(3) Let a and b be a pair of variables in consideration. Find their joint state transition matrix τ^{ab}. Here, τ^{ab}_{ijk} will denote the probability of transitioning from state i in state sequence a and state j in state sequence b to symbol k in symbolic sequence b;
(4) Short subsequences of symbols obtained from training sequences are considered, and then we evaluate T^{ab} $\forall a, b$ for each short subsequence using Eq. (10); and
(5) Real values of T^{ab} are converted to binary values by finding a maximum margin hyperplane. In this case, we would obtain a threshold and values below it would be assigned 0.
(6) A fixed time window W is considered when selecting short subsequences. The window moves over the entire time series. Each input pattern for training an RBM is obtained as one short subsequence (time window) from the binary pattern P^{ab} $\forall a, b$ (computed in the previous step). A trained RBM assigns high probability or low free energy to training patterns.

5 Experimental Setup for Validation

To demonstrate that our framework can successfully identify and isolate anomalous events, we focus on an experimental setup with a single arm robotic manipulator. Robotic manipulators have formed the backbone for several supply chain based industries and performance monitoring of such systems have not received a lot of interest. While physical failures occurring within a robot are relatively easy to observe and monitor, intelligent hacks meant to reduce overall productivity can be very hard to detect. Detailed motivation for choosing this practical experiment is included in the supplementary section available in the Supplemental Materials on the ASME Digital Collection. We present a case study based on multivariate time series values collected from this system which are used to discriminate anomalous modes of operation from nominal modes. This technique of anomaly detection and root cause analysis is crucial for attack resilient and safety-critical systems.

5.1 Data and Testbed. We have made use of the multivariate time series data collected from the single arm manipulator Sawyer after before and after anomalies were simulated on the system. Figure 1 shows a schematic of joint assignment and degrees-of-freedom, which we will use later to analyze the results.

We obtained data from 27 controlled variables denoting the positions, velocities, and accelerations of each of the seven joint angles (as illustrated in Fig. 1), along with the gripper linear position, velocity, acceleration and angles of the head-pan. Out of the remaining 19 variables, we include the five three-dimensional variables (i.e., position, linear twist, angular twist, wrench force, and wrench torque) for x, y, and z dimensions and one four-dimensional variable (i.e., orientation) of the end effector. From the experimental setup, we can see that there is a causal relationship between the controlled and observed variables. In Sec. 6, we discuss how we change controlled variables, simulating a cyber-attack, and show that our method can detect the change using only observed variables.

6 Experiments, Results, and Discussion

In this section we test our algorithm on three general cases of anomaly attacks on the robot, and compare performance of Spatio Temporal Pattern Network (STPN) framework as proposed in Ref. [21], transfer entropy metric based STPN (T-STPN) as proposed in Ref. [12], and our newly proposed G-STPN.

6.1 Attack Injection. Figure 4 provides a visualization of all the attack types with a time series of image frames.

Type 1: controller hack: By changing the controlled value of single or several joints, for example, joint 0, 2, 3, and 4, we simulate a controller hack situation. A simpler example would be multiplying the joint values of joints 0 and 3 by a time dependent "error factor" e_i of the following form:

$$e_i = \begin{cases} \dfrac{|c-i|}{c}, & \text{if } i \le c \\ \dfrac{|i|}{c}, & \text{otherwise } i > c \end{cases}$$

Here, c is defined as a positive constant specifically for the experiments and we let i be the time index of first c points in the acquired dataset Q. Thereafter, we multiply the values of joints 2 and 4 by the same time-dependent "error factor" e_i with a time shift

$$e_i = \begin{cases} \dfrac{|d+c-i|}{c}, & \text{if } i \le d + \dfrac{c}{2} \\ \dfrac{|i-d|}{c}, & \text{otherwise } i > d + \dfrac{c}{2} \end{cases}$$

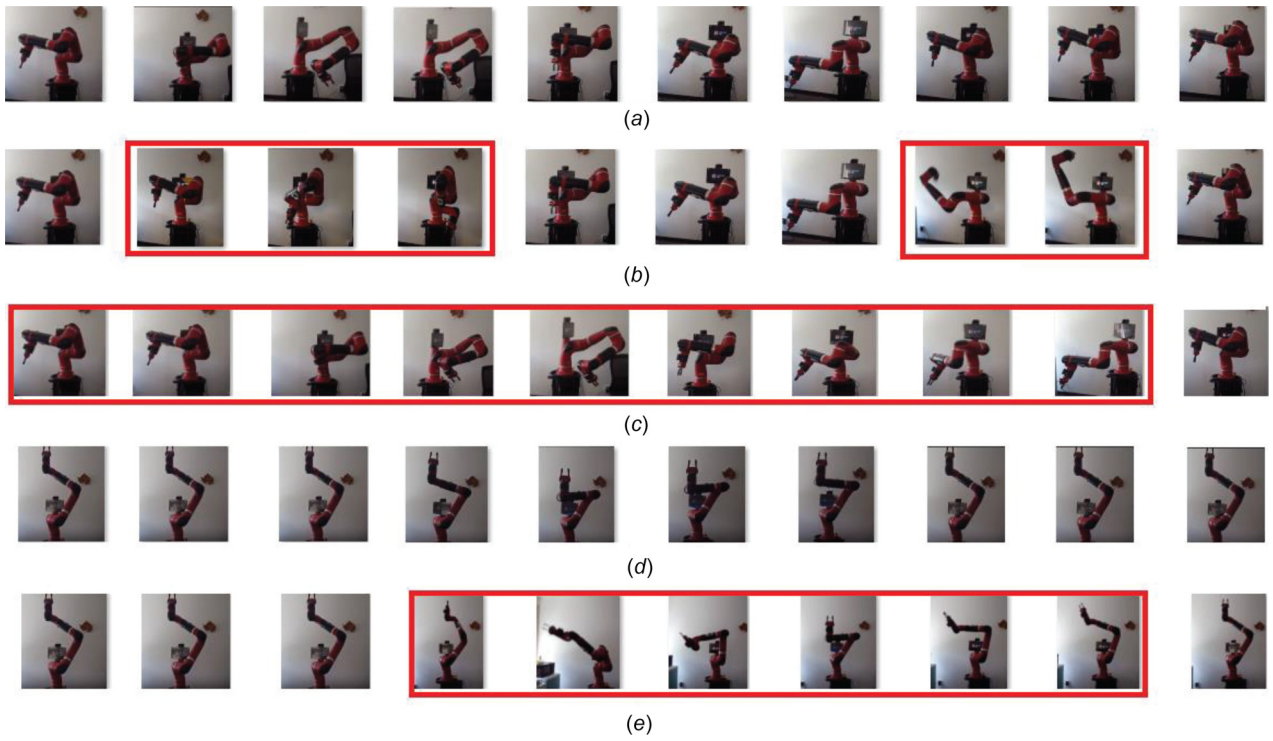

Fig. 4 Time frames showing the task execution in nominal and anomalous conditions [12]: (*a*) defined nominal operation of the robot in 10 frames, (*b*) anomalous operation of the robot due to controller hack, (*c*) anomalous operation of the robot due to communication delay, (*d*) nominal trajectory of the robot by following commands from moveit, and (*e*) anomalous trajectory of the robot following anomalous commands from moveit because of imaginary obstacle

Again, c and d positive constants are chosen to be kept fixed in the experiments and the time index in dataset Q between indices d and $d+c$ is denoted by i.

Type 2: communication delay: In order to simulate communication delay, the first c joint 3 control values are set equal to the first data point value (i.e., we apply zero order hold). After the cth data point the following equation gives the value for joint 3

$$Q(i,3) = \begin{cases} Q(1,3), & \text{if } i \le c \\ Q(i-c,3) & \text{otherwise } i > c \end{cases}$$

All other joint values in the dataset Q remain unchanged.

Type 3: trajectory manipulation: For simulating intelligent attacks of type (3), we use the MoveIt! inverse kinematics library with visualization in Rviz. The library is typically used to solve the inverse kinematics problem of the desired minimum path length trajectory for a given starting and ending end effector position and orientation. Nominal training data is obtained by collecting commanded joint angles and end effector positions for a given starting and ending position and orientation of end effector. We place an imaginary obstacle in between the shortest trajectory path and force the inverse kinematics based trajectory planner to recalculate a longer optimal path. In this, we introduce trajectory manipulation anomaly. The graphical structure learned by our algorithm for this case is provided in supplementary section available in the Supplemental Materials on the ASME Digital Collection. However, keeping in mind the complexity of the problem in establishing exact ground truth as a baseline for our algorithm, detailed analysis of this special case of anomaly is left as future work.

6.2 Spatiotemporal Anomaly Detection. The anomaly detection algorithm proposed here is able to detect causes of system anomaly on both spatial and temporal scales. It can provide explanations of failure at each individual subsystem as well as the explanations for failure because of anomalous interactions among subsystems. The results essentially present us with a three-dimensional graph with nodes, time points, and anomaly scores as x, y, and z axis, respectively. To simplify the interpretation, we take the average of all the anomaly scores along time to produce the following explanation: *node anomaly scores averaged over time*, presented in Figs. 5 and 6 for STPN, T-STPN, and G-STPN frameworks. The anomaly score values along the y-axis are relative importance given to events along time and nodes, and is thus scale independent across different algorithms (only relative differences matter).

The time period of operation of the robot is roughly 35 s. This work is repeatedly performed and recorded as nominal data for around 12 min. The frequency of data collection is 100 datapoints per second. The Window size for extracting importance metrics is 2000 points (=20 s), the window stride length is 10 (=0.1 s). During inference in root cause analysis (RCA) technique, a collection of 50 windows are considered together (=5 s resolution for anomaly detection). We call each of these collection of 50 windows an instant.

For *communication delay*, joint 4 was programmed to have the time lag anomaly and the time averaged node score (T-STPN) across the nodes shows that node 4 joint position is the most anomalous. However, STPN attributes majority of root causes to the (observed) end effector variables. G-STPN attributes majority of causes to joint 6 acceleration and joint 1 positions. This is not surprising as both of them are equally distant from joint 3 and due to operating mechanism are heavily influenced by perturbations in joint 3 operation.

For *controller hack*, joint 1, 3, 4, 5 controllers had been programmed for anomaly. In the time averaged node score plot STPN detects acceleration anomaly of joint 6, T-STPN detects velocity anomaly of joint 4, and G-STPN detects velocity anomaly of joint 3 and 4 with high relative weights. It is evident by comparison, that for these cases, the results offered by G-STPN is more stable and accurate over the results produced by STPN and T-STPN.

6.3 Performance Analysis. In Figs. 7–9 and 10–12, we can see the raw data with free energy plots derived from the RBM

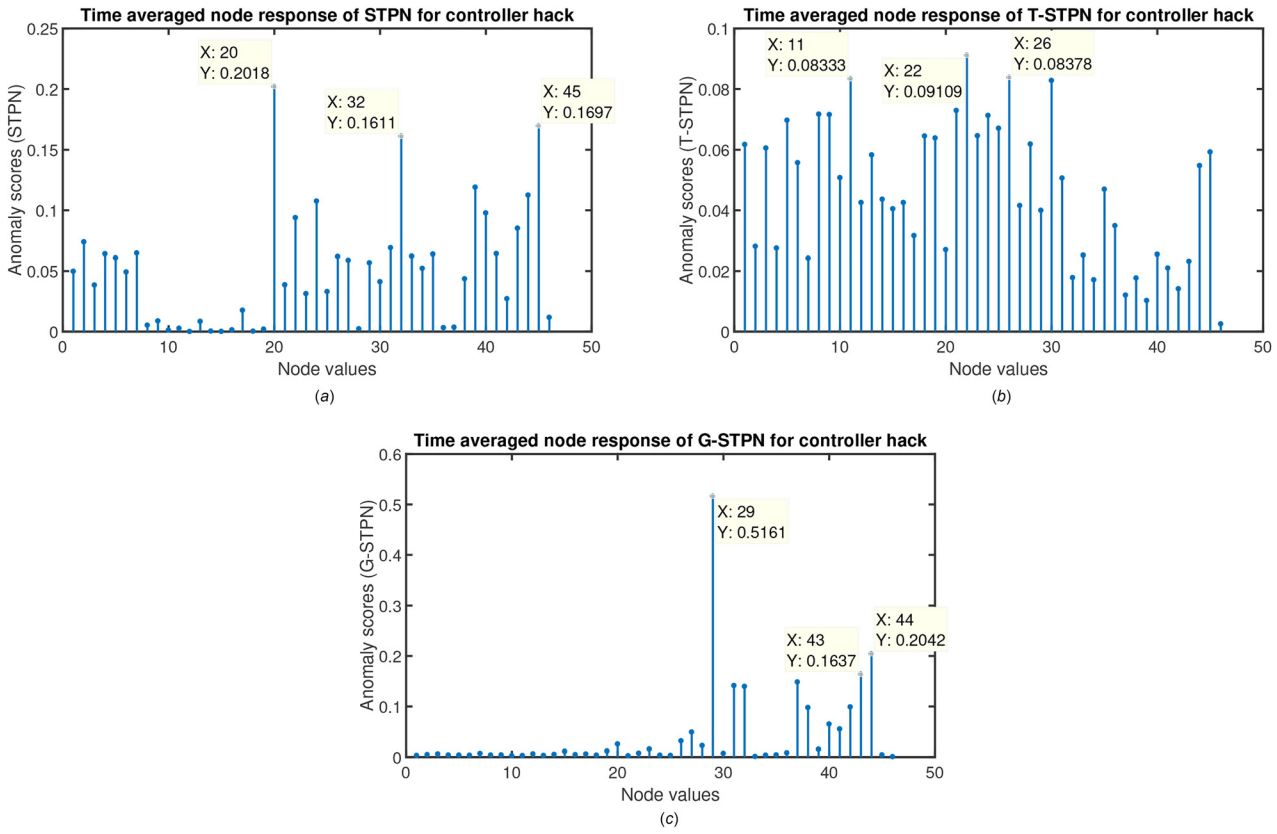

Fig. 5 (left to right) STPN, G-STPN, and T-STPN (controller hack)-anomaly scores across nodes. G-STPN can produce much sharper peaks pointing out anomalous nodes with greater certainty.

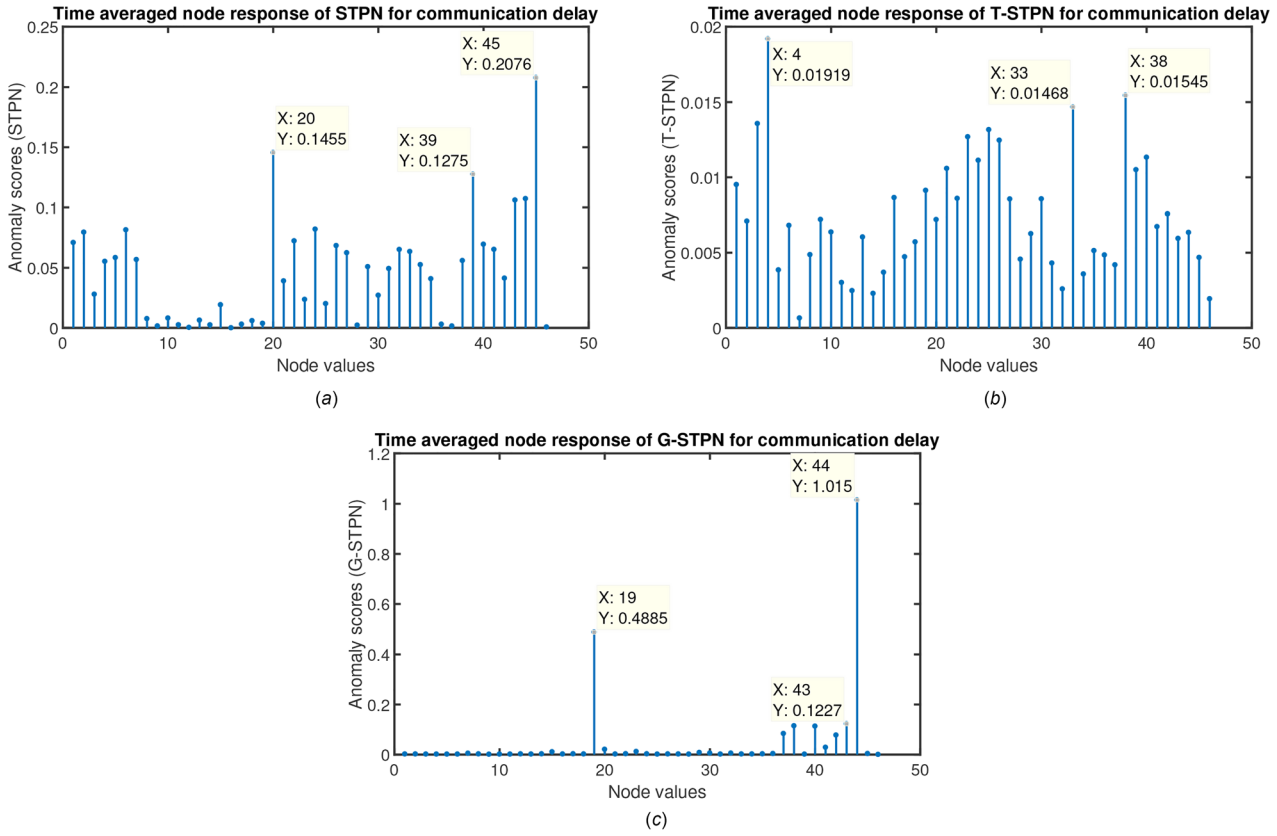

Fig. 6 (left to right) STPN, G-STPN, and T-STPN (communication delay)-anomaly scores across nodes. G-STPN can produce much sharper peaks pointing out anomalous nodes with greater certainty.

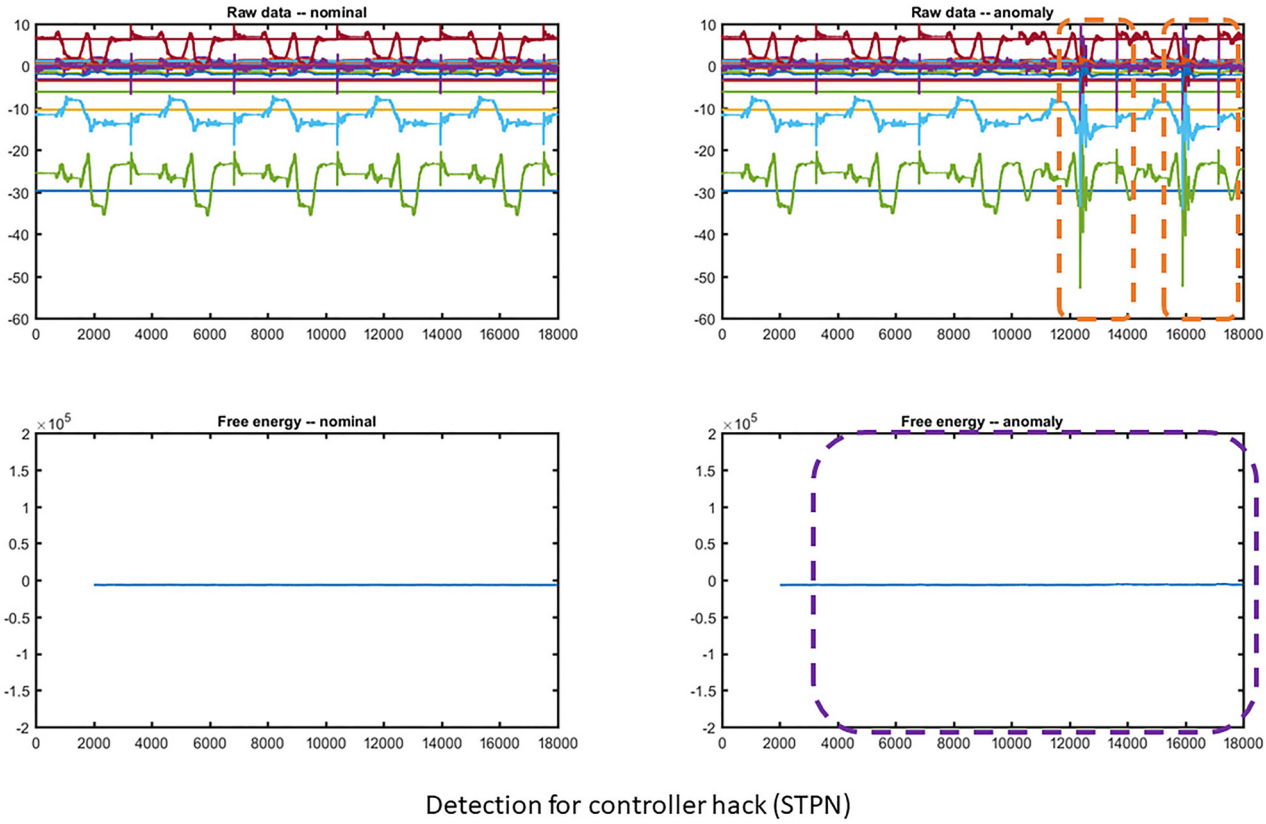

Fig. 7 Raw data and free energy of RBM under nominal and anomalous conditions for the controller hack (STPN)—note that the anomalous free energy distribution is flat and is not any higher than the nominal case

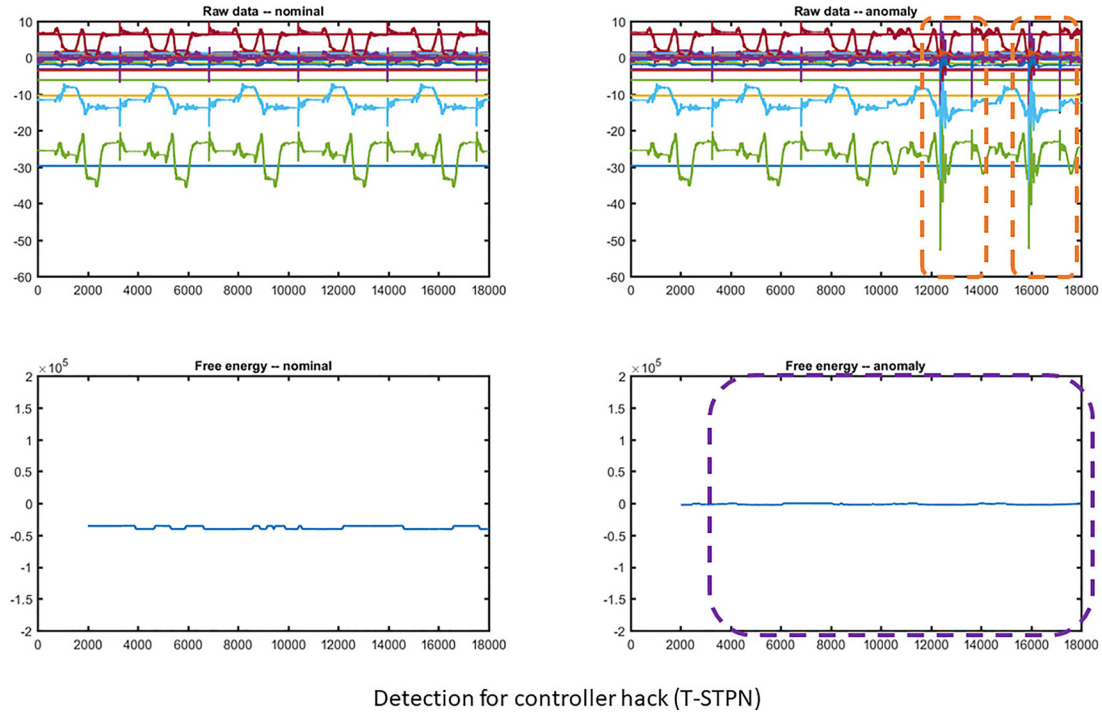

Fig. 8 Raw data and free energy of RBM under nominal and anomalous conditions for the controller hack (T-STPN)—note that the anomalous free energy distribution is flat but sits higher than the nominal case denoting that the entire series is captured as anomaly

trained on nominal data. In the case of raw data, the time series of every variable is combined and showed in one plot. For all of three cases, presence of anomaly in raw data is manifested as an increase in free energy. Anomaly has been introduced at 15,000 s, after which the robot operates in an anomalous operation cycle till the end. Free energy has been calculated on the entire symbolic time series which includes events before and after the anomaly. It is seen that just based on free energies, STPN and T-STPN

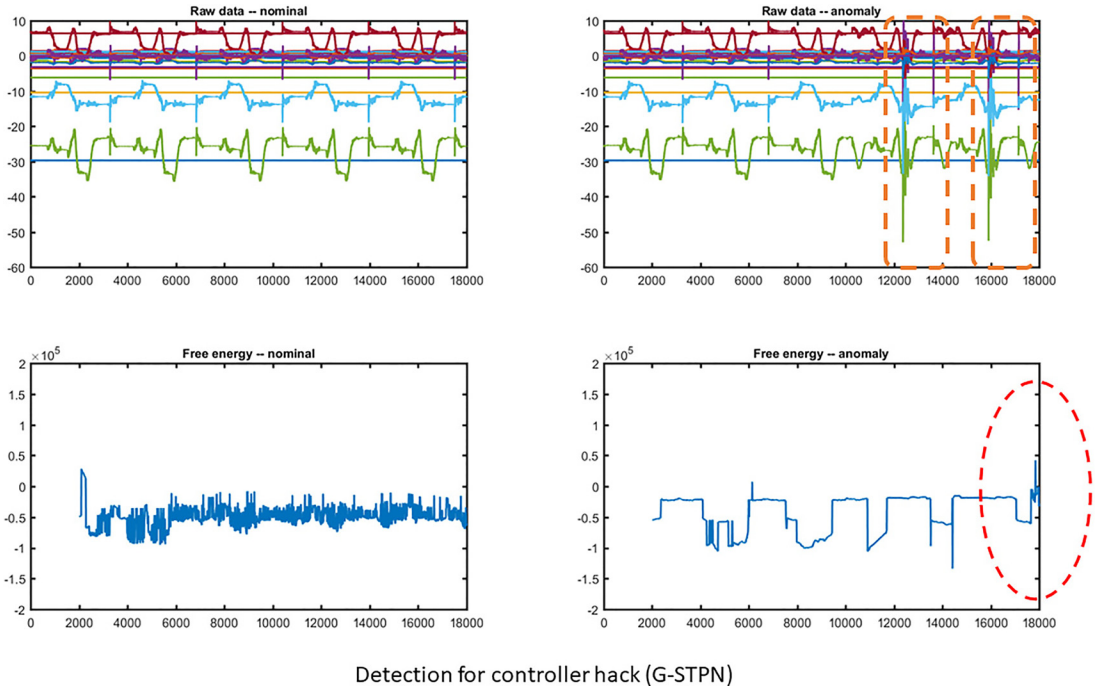

Fig. 9 Raw data and free energy of RBM under nominal and anomalous conditions for the controller hack (G-STPN)—note that it is able to precisely detect the two anomalous events (marked with orange squares) as the spike (inside red circle), albeit in a lagged fashion

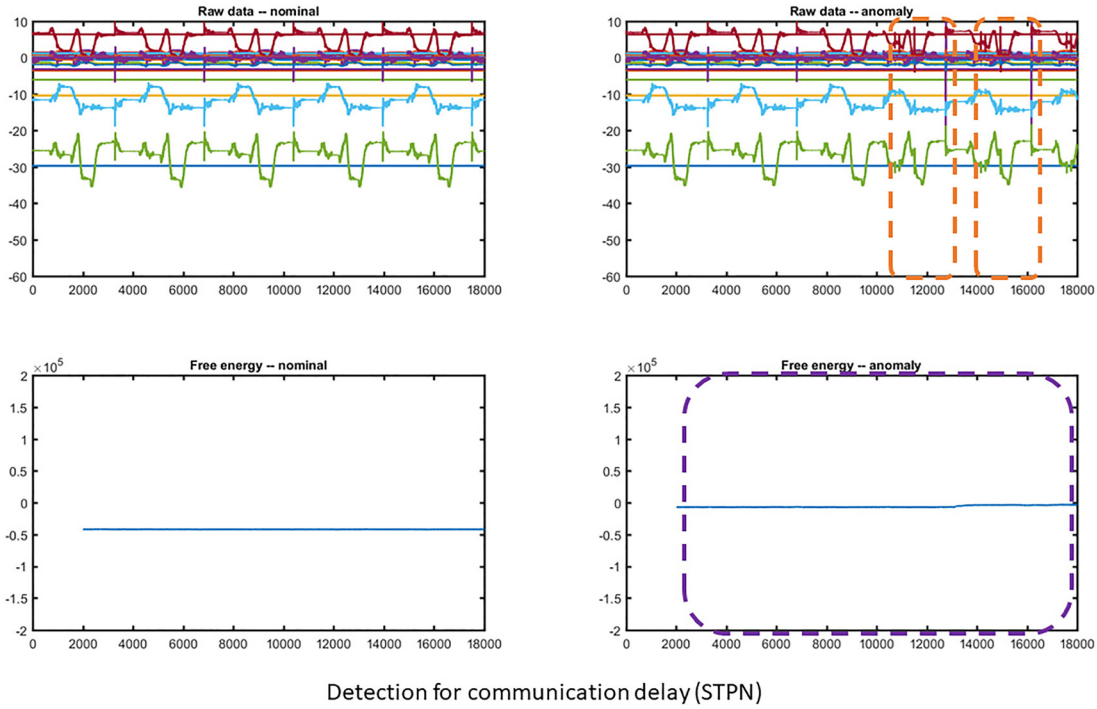

Fig. 10 Raw data and free energy of RBM under nominal and anomalous conditions for the communication delay (STPN)—note that the anomalous free energy distribution is flat but is higher than the nominal case

frameworks are not able to indicate introduction of anomalies based on jumps in free energy (a fact not investigated in Ref. [12]). However G-STPN shows sudden jumps at 16,000–18,000 time window for controller hack, and the time windows 12,000–14,000; 16,000–18,000 for communication delays.

This change in nature of the free energy can also be investigated by various state of the art change detection algorithms to trigger defense mechanisms. This is in line with several state of the art data driven techniques that tend to characterize anomalous events as departures from known probability distributions in the training data [22–24].

Figures 5 and 6 shows the root causes corresponding to controller hack and communication delay, respectively. In the case of controller hack, anomalies are injected to joints 0, 2, 3, and 4. Along with those, joint 5 is also considered as anomaly because it is directly connected to the 2, 3, 4 chains. By using the proposed

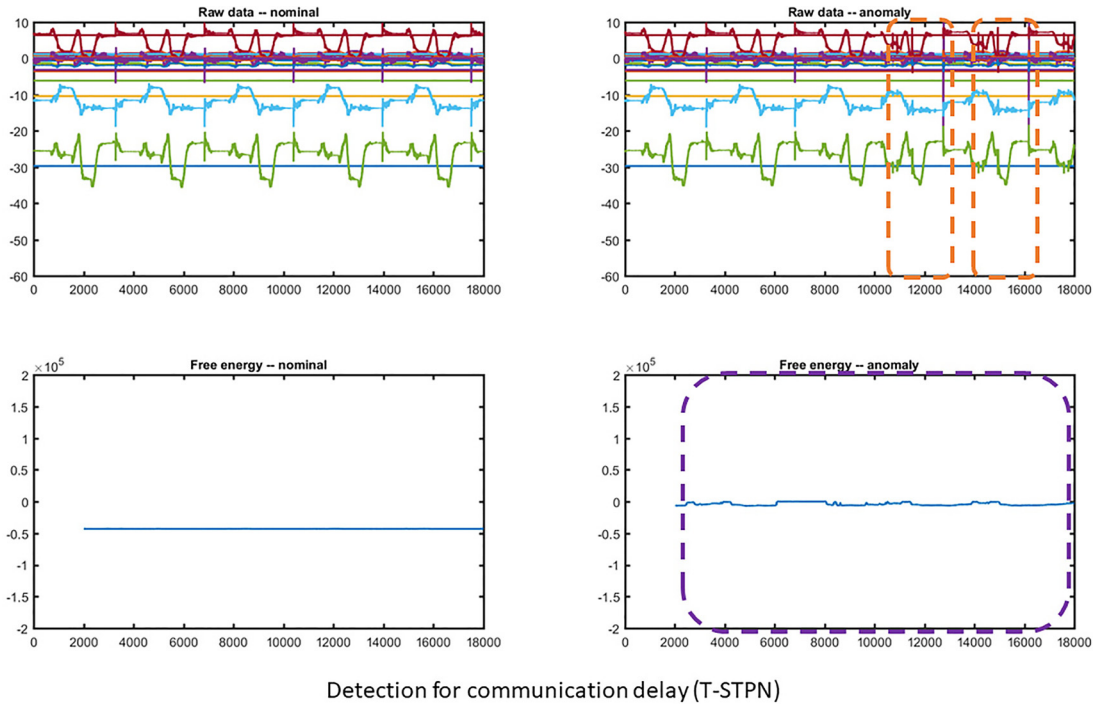

Fig. 11 Raw data and free energy of RBM under nominal and anomalous conditions for the communication delay (T-STPN)—note that the anomalous free energy distribution is almost flat and is higher than the nominal case

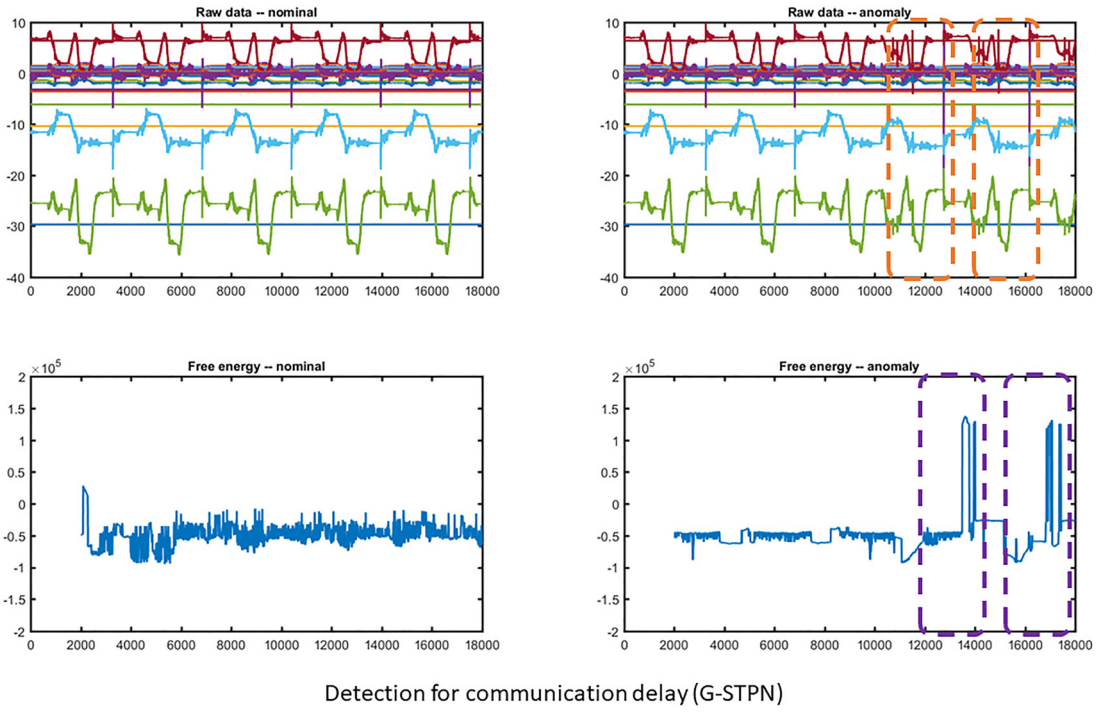

Fig. 12 Raw data and free energy of RBM under nominal and anomalous conditions for the communication delay (G-STPN)—note that it precisely detects the anomalies (orange squares) as the spikes inside purple squares

approach, except joint 5, the rest of four joints can be isolated correctly. As joint 5 is next to joint 4 the isolation of joint 4 may negatively affect the isolation of joint 5.

For the communication delay, it can be observed that the anomaly is injected to joint 3. Although eventually using the proposed RCA method enables us to isolate three joints, i.e., joints 0, 3, and 4, where anomalies are detected, joint 3 can be correctly detected to help operators locate the attacks.

From the figures, we can observe that for the case of communication delay, anomalies are identified to be injected at joint 3. After using the proposed RCA technique, we can isolate three joints, i.e., joints 0, 3, and 4.

Joint 4 is also isolated in this case as it is close to joint 3 while for joint 0, the reason may be attributed to the robot's dynamics which is not analyzed in detail in this work. For the case of trajectory manipulation, we can know that when a block is

placed in the path from starting point to end, joints can be observed to move for avoiding the block. However, it is difficult to determine the ground truth in terms of joints as the trajectories involve most of the joints. Also, based on the proposed algorithm, results show that most of joints are involved so the isolation of joints is not provided for the root-cause analysis in the trajectory manipulation case.

7 Conclusions and Future Work

In this paper, we thoroughly analyze a family of techniques for anomaly detection and root cause analysis in complex cyber-physical-systems, namely, STPN, and its variants Transfer entropy based T-STPN, and a nonlinear granger causality based G-STPN (which we propose in this paper). All of the techniques leverage abstraction of time series dynamics through SDF. However, by investigating variations of Granger causality (nonlinear version proposed by us), between each pair of time series within the multivariate time series, causes of anomaly can be easily found out. We provide a detailed mathematical formulation, perform a simulated Monte Carlo case study for classical time series models studied in literature for causality, and finally apply our framework to a real world scenario of detection cyber-attacks on an industrial manipulator. The key technical contributions are summarized below:

(1) The proposed technique of data driven anomaly detection can be used in performance monitoring of large scale industrial processes and check for intruders and potential hackers that may significantly lower productivity.
(2) Proposed metric of causality detection can be used as a replacement for transfer entropy and can be used to detect causal interactions in multivariate time series data.
(3) Proposed framework when integrated with root cause analysis approach is more sensitive in detecting individual anomalous variables in a process than earlier STPN, T-STPN frameworks and hence can be used as an improvement upon existing root cause analysis techniques.

Some limitations of our proposed G-STPN framework are:

(1) Our framework relies upon abstracting continuous multivariate time series data using the technique of SDF, which is not guaranteed to capture the complete dynamics of a continuous time series because it heavily relies upon the partitioning scheme [25].
(2) Computation load and data requirement may grow dramatically with larger embedding dimension (k-refer to Fig. 3) and larger alphabet size Σ (refer Sec. 2). Both of these variables need to be increased in order to capture increasingly complex patterns.
(3) Statistical stationarity is needed to reliably estimate the transition probabilities.

Possible future research directions are:

(1) We propose to evaluate joint and self-symbol generation matrices corresponding to pairs of time series. A method that would use our Dirichlet priors of joint symbol generation matrix and self-symbol generation matrix to arrive at and improve estimates of a new nonlinear Granger causality metric is left as future work.
(2) Establishing theoretical bounds on optimal length of window and embedding depth required for capturing time series dynamics using SDF.
(3) On the implementation side, a rigorous complexity analysis and considerations when implementing on cheap real time systems.
(4) Online interception of cyber-physical attacks and incorporating attack patterns to prevent future failures is a potential future research direction. This requires understanding of temporal attack propagation characteristics.

Acknowledgment

This work was supported in part by NSF Grants CNS-1845969 (CAREER), CNS-1932033, NSF/USDA NIFA Grant 2017-67007-26151, and AFOSR YIP Grant FA9550-17-1-0220.

Funding Data

- National Science Foundation (Funder ID: 10.13039/100000001).

References

[1] Liu, C., Akintayo, A., Jiang, Z., Henze, G. P., and Sarkar, S., 2018, "Multivariate Exploration of Non-Intrusive Load Monitoring Via Spatiotemporal Pattern Network," Appl. Energy, **211**, pp. 1106–1122.
[2] Liu, C., Ghosal, S., Jiang, Z., and Sarkar, S., 2016, "An Unsupervised Spatiotemporal Graphical Modeling Approach to Anomaly Detection in Distributed Cps," Proceedings of the Seventh International Conference on Cyber-Physical Systems, Vienna, Austria, Apr.
[3] Dunbabin, M., and Marques, L., 2012, "Robots for Environmental Monitoring: Significant Advancements and Applications," IEEE Rob. Autom. Mag., **19**(1), pp. 24–39.
[4] Darianian, M., and Michael, M. P., 2008, "Smart Home Mobile RFID-Based Internet-of-Things Systems and Services," International Conference on Advanced Computer Theory and Engineering (ICACTE'08), Phuket, Thailand, Dec. 20–22, pp. 116–120.
[5] Jiang, Z., Liu, C., Akintayo, A., Henze, G. P., and Sarkar, S., 2017, "Energy Prediction Using Spatiotemporal Pattern Networks," Appl. Energy, **206**, pp. 1022–1039.
[6] Granger, C. W., 1988, "Causality, Cointegration, and Control," J. Econ. Dyn. Control, **12**(2–3), pp. 551–559.
[7] Dimpfl, T., and Peter, F. J., 2013, "Using Transfer Entropy to Measure Information Flows Between Financial Markets," Stud. Nonlinear Dyn. Econometrics, **17**(1), pp. 85–102.
[8] Vicente, R., Wibral, M., Lindner, M., and Pipa, G., 2011, "Transfer Entropy–A Model-Free Measure of Effective Connectivity for the Neurosciences," J. Comput. Neuroscience, **30**(1), pp. 45–67.
[9] Ver Steeg, G., and Galstyan, A., 2012, "Information Transfer in Social Media," Proceedings of the 21st International Conference on World Wide Web, Lyon, France, pp. 509–518.
[10] Gupta, S., and Ray, A., 2007, "Symbolic Dynamic Filtering for Data-Driven Pattern Recognition," Pattern Recognit.: Theory Appl., **2**, pp. 17–71.
[11] Liu, C., Zhao, M., Sharma, A., and Sarkar, S., 2019, "Traffic Dynamics Exploration and Incident Detection Using Spatiotemporal Graphical Modeling," J. Big Data Anal. Transp., **1**(1), pp. 37–55.
[12] Saha, H., Liu, C., Jiang, Z., and Sarkar, S., 2018, "Exploring Granger Causality in Dynamical Systems Modeling and Performance Monitoring," IEEE Conference on Decision and Control (CDC), Miami, FL, Dec. 17–19, pp. 2537–2542.
[13] Liu, C., Gong, Y., Laflamme, S., Phares, B., and Sarkar, S., 2017, "Bridge Damage Detection Using Spatiotemporal Patterns Extracted From Dense Sensor Network," Meas. Sci. Technol., **28**(1), p. 014011.
[14] Han, T., Liu, C., Wu, L., Sarkar, S., and Jiang, D., 2019, "An Adaptive Spatiotemporal Feature Learning Approach for Fault Diagnosis in Complex Systems," Mech. Syst. Signal Process., **117**, pp. 170–187.
[15] Tan, S. Y., Saha, H., Florita, A. R., Henze, G. P., and Sarkar, S., 2019, "A Flexible Framework for Building Occupancy Detection Using Spatiotemporal Pattern Networks," National Renewable Energy Lab. (NREL), Golden, CO, Report No. NREL/CP-5D00-73359.
[16] Liu, C.-L., Hsaio, W.-H., and Tu, Y.-C., 2019, "Time Series Classification With Multivariate Convolutional Neural Network," IEEE Trans. Ind. Electron., **66**(6), pp. 4788–4797.
[17] Saha, H., Venkataraman, V., Speranzon, A., and Sarkar, S., 2019, "A Perspective on Multi-Agent Communication for Information Fusion," arXiv preprint arXiv:1911.03743.
[18] Saha, H., Tan, S. Y., Jiang, Z., and Sarkar, S., 2019, "Learning State Switching for Multi Sensor Integration," Indian Control Conference (ICC), Hyderabad, India.
[19] Sarkar, S., Mukherjee, K., Sarkar, S., and Ray, A., 2013, "Symbolic Dynamic Analysis of Transient Time Series for Fault Detection in Gas Turbine Engines," ASME J. Dyn. Syst. Meas. Control, **135**(1), p. 014506.
[20] Liu, C., Ghosal, S., Jiang, Z., and Sarkar, S., 2017, "An Unsupervised Anomaly Detection Approach Using Energy-Based Spatiotemporal Graphical Modeling," Cyber-Phys. Syst., **3**(1–4), pp. 66–102.
[21] Liu, C., Lore, K. G., and Sarkar, S., 2017, "Data-Driven Root-Cause Analysis for Distributed System Anomalies," IEEE 56th Annual Conference on Decision and Control (CDC), Melbourne, Australia, Dec. 12–15, pp. 5745–5750.
[22] Dorj, E., Chen, C., and Pecht, M., 2013, "A Bayesian Hidden Markov Model-Based Approach for Anomaly Detection in Electronic Systems," IEEE Aerospace Conference, Big Sky, MT, Mar. 2–9, pp. 1–10.
[23] Fiore, U., Palmieri, F., Castiglione, A., and De Santis, A., 2013, "Network Anomaly Detection With the Restricted Boltzmann Machine," Neurocomputing, **122**, pp. 13–23.
[24] Patcha, A., and Park, J.-M., 2007, "An Overview of Anomaly Detection Techniques: Existing Solutions and Latest Technological Trends," Comput. Networks, **51**(12), pp. 3448–3470.
[25] Bollt, E. M., Stanford, T., Lai, Y.-C., and Życzkowski, K., 2001, "What Symbolic Dynamics Do We Get With a Misplaced Partition?: On the Validity of Threshold Crossings Analysis of Chaotic Time-Series," Phys. D, **154**(3–4), pp. 259–286.