

State Variable Effects in Graphical Event Models

Debarun Bhattacharjya, Dharmashankar Subramanian and Tian Gao

Research AI, IBM T. J. Watson Research Center

{debarunb, dharmash, tgao}@us.ibm.com

Abstract

Many real-world domains involve co-evolving relationships between events, such as meals and exercise, and time-varying random variables, such as a patient’s blood glucose levels. In this paper, we propose a general framework for modeling joint temporal dynamics involving continuous time transitions of discrete state variables and irregular arrivals of events over the timeline. We show how conditional Markov processes (as represented by continuous time Bayesian networks) and multivariate point processes (as represented by graphical event models) are among various processes that are covered by the framework. We introduce and compare two simple and interpretable yet practical joint models within the framework with relevant baselines on simulated and real-world datasets, using a graph search algorithm for learning. The experiments highlight the importance of jointly modeling event arrivals and state variable transitions to better fit joint temporal datasets, and the framework opens up possibilities for models involving even more complex dynamics whenever suitable.

1 Introduction & Related Work

Several domains involve an underlying causal mechanism with co-evolving dynamics where the states of uncertain variables and their transitions influence arrivals of events, and vice versa. We make a semantic distinction between two kinds of variables – **event labels** and **state** (or system) **variables**. Event labels are categories of events, occurring instantaneously and often irregularly on the timeline. In contrast, state variables are variables which always exist in some state, with potential state changes observed at irregularly timed transitions. For instance, meals and exercise could be viewed as event labels, whereas a patient’s blood glucose level (say with states for low, medium and high) is a state variable.

There is substantial literature on graphical models for dynamic systems. A useful distinction between the various avenues of research is based on whether observations are in discrete or continuous time. Dynamic Bayesian networks [Dean and Kanazawa, 1989; Murphy, 2002] are discrete-time models that extend Bayesian networks [Pearl, 2014] by jointly

representing a set of discrete variables at regular epochs. Time series data are continuous valued measurements, typically also observed at regular epochs, and can be represented by Granger causal graphs [Eichler, 1999]. A stream of graphical modeling work for time series however also attempts to handle irregular observations such as through using additional labels for missing data [Kolar *et al.*, 2010; Zhou *et al.*, 2010] or through a sampling-rate-agnostic learning approach [Plis *et al.*, 2015].

There is a vast body of burgeoning research on temporal processes for modeling multivariate streams of events, spanning parametric approaches and neural network architectures [Rajaram *et al.*, 2005; Simma and Jordan, 2010; Weiss and Page, 2013; Goulding *et al.*, 2016; Du *et al.*, 2016; Xiao *et al.*, 2017; Gao *et al.*, 2020]. Such models use a *marked point process* [Cox and Lewis, 1972] to capture the continuous-time dynamics in event streams, varying in assumptions and parametrization around the historical dependencies between the various types of events. Graphical event models (GEM) [Didelez, 2008; Meek, 2014] are a high-level framework for representing marked point processes in a graphical form; the framework subsumes a large class of temporal models for events.

Continuous time Bayesian networks (CTBN) [Nodelman *et al.*, 2002] are a related model that represent conditional Markov processes where a joint set of variables make state transitions in an inter-dependent fashion. Some recent work on event-driven CTBNs (ECTBN) provides an extension enabling the effect of external events on state variables [Bhattacharjya *et al.*, 2019; Bhattacharjya *et al.*, 2020]. However, this model is limited in handling joint dynamics – it retains Markovian dependence between state variables and does not permit them to affect event arrivals, as can be common in numerous applications.

Our main contribution in this paper is a novel unifying framework that we refer to as a state variable graphical event model (SVGEM); this family of models jointly captures the co-evolution of state variables and event label arrivals in continuous time. The framework extends beyond the Markov assumptions in CTBNs/ECTBNs, and allows for state variables and event labels to dynamically influence each other, potentially in complex ways. It is therefore capable of representing a wide variety of process models. We theoretically demonstrate the generality of the SVGEM framework, showing how

it encapsulates many models and existing frameworks including GEMs. Through experiments on simulated and real-world benchmark datasets, two specific models from the proposed framework are shown to outperform various baselines, highlighting the promising modeling power of the high-level framework.

2 Model Formulation

2.1 Basic Notation & Terminology

Variables & Data. We distinguish between two kinds of variables. **Event labels** are denoted $\mathcal{E} = \{E_j\}_{j=1}^J$. We assume there is data about events occurring over time, $D_{\mathcal{E}} = (t_k, e_k)_{k=1}^{N_{\mathcal{E}}}$, where t_k are ordered time stamps between initial time $t_0 = 0$ to the end time T , and e_k belong to the event label set \mathcal{E} . Discrete **state variables** are denoted $\mathcal{X} = \{X_i\}_{i=1}^I$. Let $\text{Val}(X_i)$ be the domain of variable X_i and $\text{Val}(\mathbf{X})$ the domain of the set of state variables \mathbf{X} . Data about each variable is of the form of state transitions, $D_{X_i} = (t_k, x_k)_{k=0}^{N_i}$ up to end time T , where the state at time t_0 is the initial state and $x_{k+1} \neq x_k \forall k, x_k \in \text{Val}(X_i)$. Data for all state variables taken together is denoted $D_{\mathcal{X}} = \bigcup_{X \in \mathcal{X}} D_X$.

Definition 1. An **occurrence** refers to either an event label arrival or a state variable transition, i.e. an occurrence is of the form (t_k, z_k) where z_k is either an event label, $e_k \in \mathcal{E}$, or a state of a state variable, $x_k \in \text{Val}(X)$, $X \in \mathcal{X}$. Here z_k is the **occurrence label**.

We use $D_{\mathbf{Z}}$ to denote data of all occurrences pertaining to both kinds of variables in \mathbf{Z} , and $D = D_{\mathcal{E}, \mathcal{X}}$ to denote data for all occurrences of all model variables.

Historical Dependence. The dynamics of event label arrivals and state variable transitions are driven by historical dependencies. We use $h(\cdot)$ to denote historical occurrences of either type of variable, i.e. event arrivals or state variable transitions. Consider variables $\mathbf{Z} = \{\mathbf{Z}^{\mathcal{E}}, \mathbf{Z}^{\mathcal{X}}\}$, where $\mathbf{Z}^{\mathcal{E}}$ and $\mathbf{Z}^{\mathcal{X}}$ are sets of event labels and state variables respectively. Then $h_{\mathbf{Z}^{\mathcal{E}}}(t) = \{(t_k, e_k) \in D_{\mathbf{Z}^{\mathcal{E}}} : t_k < t\}$ represents the history of event arrivals in the set $\mathbf{Z}^{\mathcal{E}} \subset \mathcal{E}$ until time t , and $h_{\mathbf{Z}^{\mathcal{X}}}(t) = \{(t_k, x_k) \in D_{\mathbf{Z}^{\mathcal{X}}} : t_k < t\}$ represents the history of state variable transitions associated with the set $\mathbf{Z}^{\mathcal{X}} \subset \mathcal{X}$ until time t . The combined history is $h_{\mathbf{Z}}(t) = h_{\mathbf{Z}^{\mathcal{E}}}(t) \cup h_{\mathbf{Z}^{\mathcal{X}}}(t)$. $h_{\mathbf{Z}}^*(t)$ refers to the most recent occurrence label in the history $h_{\mathbf{Z}}(t)$.

2.2 The SVGEM Framework

We now have the notation and terminology required to formalize the general SVGEM framework:

Definition 2. An SVGEM \mathcal{M} includes:

- A directed (possibly cyclic) graph \mathcal{G} where:
 - Every event label $E \in \mathcal{E}$ has parents $\mathbf{U}_E = \{\mathbf{U}_E^{\mathcal{E}}, \mathbf{U}_E^{\mathcal{X}}\}$, where $\mathbf{U}_E^{\mathcal{E}} \subseteq \mathcal{E}$ and $\mathbf{U}_E^{\mathcal{X}} \subseteq \mathcal{X}$ are event label and state variable parents respectively
 - Every state variable $X \in \mathcal{X}$ has parents $\mathbf{U}_X = \{\mathbf{U}_X^{\mathcal{E}}, \mathbf{U}_X^{\mathcal{X}}\}$, where $\mathbf{U}_X^{\mathcal{E}} \subseteq \mathcal{E}$ and $\mathbf{U}_X^{\mathcal{X}} \subseteq \{\mathcal{X} \setminus X\}$ are event label and state variable parents respectively
- An initial distribution $\mathbf{P}_{\mathcal{X}}^0$ over state variables

- Conditional intensity rate parameters Λ as follows:

- Every event label $E \in \mathcal{E}$ occurs with rate $\lambda_{E|h_{\mathbf{U}_E}}(t)$ at time t , where $h_{\mathbf{U}_E}(t)$ denotes the history of all occurrences in parent set \mathbf{U}_E ; each set is denoted Λ_E
- Every transition s, s' for every state variable $X \in \mathcal{X}$, $s, s' \in \text{Val}(X)$; $s \neq s'$, occurs with rate $\lambda_{X|h_{\mathbf{U}_X}}^{s, s'}(t)$ at time t when $h_X^*(t) = s$, and 0 otherwise, where $h_{\mathbf{U}_X}(t)$ denotes the history of all occurrences in parent set \mathbf{U}_X ; each set is denoted Λ_X

An SVGEM captures the joint dynamics of all occurrences based on an underlying graph that specifies the causal factors influencing each variable. Fig. 1(a) shows an illustrative SVGEM graph involving 2 state variables and 3 event labels. We highlight that the definition imposes an important constraint: a state variable cannot transition from state s to s' if it is not already in state s . Also, note that an SVGEM is merely a framework or a family of models, similar to a GEM – for a fully specified generative model, more details about the general historical dependence (denoted by Λ_E and Λ_X) need to be provided, such as what is described next.

2.3 Types of Historical Dependence

We categorize various kinds of historical dependence for either type of variable. These are by no means exhaustive but cover the cases required for this article. All of these are stationary in the sense that an occurrence's rate is independent of the time at which history is considered, given the history.

- A historical dependence is **Markov** w.r.t set \mathbf{Z} if the rate at any time t depends only on $h_{\mathbf{Z}}^*(t)$, i.e. the most recent occurrence label in the history $h_{\mathbf{Z}}(t)$.
- A historical dependence is **piece-wise constant (PC)** w.r.t set \mathbf{Z} if there is a fully specified mapping from the history $h_{\mathbf{Z}}(t)$ at any time t into some discrete state space Σ [Gunawardana *et al.*, 2011]. Such a mapping induces the conditional intensity rate for a variable to be piece-wise constant over the timeline for any dataset. Fig. 2 (Top) illustrates this by displaying the conditional intensity rate profile for an event label on an example dataset.
- A historical dependence is **proximal** w.r.t event label set $\mathbf{Z}^{\mathcal{E}}$ if the rate at any time t depends only on whether or not each node in $\mathbf{Z}^{\mathcal{E}}$ has occurred in a preceding time window before t [Bhattacharjya *et al.*, 2018]. Proximality w.r.t state variables $\mathbf{Z}^{\mathcal{X}}$ needs further clarification, and is deferred to later in the article. We note that this is a special but important case of PC historical dependence.
- A historical dependence is **restricted** by a subset \mathbf{W} of set \mathbf{Z} if the rate at any time t is 0 when the most recent occurrence label $h_{\mathbf{Z}}^*(t)$ belongs to \mathbf{W} . Thus, if the most recent occurrence is from \mathbf{W} , the occurrence under consideration is impossible.

2.4 Equivalent GEM for an SVGEM

We define an expanded GEM, where each state variable transition is treated as an event, that can represent the dynamic process for an underlying SVGEM. Recall that nodes $E \in \mathcal{E}$

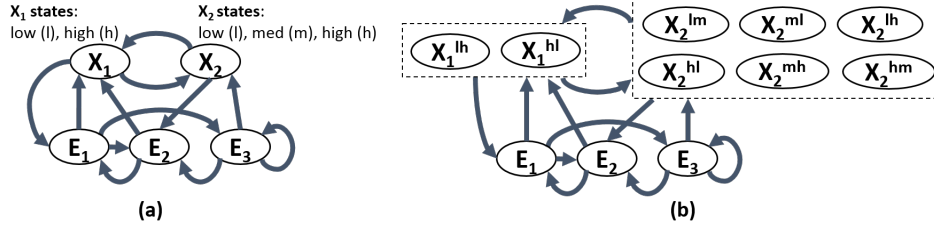


Figure 1: (a) An illustrative SVGEM; (b) The eqv. (expanded) GEM for the SVGEM in (a) where each state variable transition is represented as an event label. Each box indicates a fully connected graph where all incoming and outgoing arcs visit all nodes within the box.

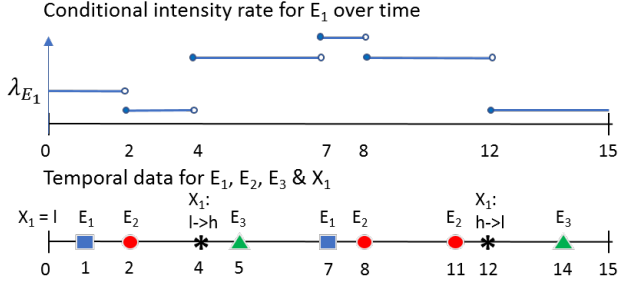


Figure 2: Bottom: Stream of E_1 (square), E_2 (circle) and E_3 (triangle) events, and transitions for state variable X_1 (marked by an asterisk); Top: Illustrative conditional intensity rate over time for event label E_1 in Fig. 1(a), assumed to be Markov w.r.t X_1 and proximal w.r.t E_2 over a time window of 5 days.

and $X \in \mathcal{X}$ in the original SVGEM have parents $\mathbf{U}_E = \{\mathbf{U}_E^\mathcal{E}, \mathbf{U}_E^\mathcal{X}\}$ and $\mathbf{U}_X = \{\mathbf{U}_X^\mathcal{E}, \mathbf{U}_X^\mathcal{X}\}$ respectively. The expanded GEM that is constructed only includes event labels, and the parents for a node E in this GEM are denoted \mathbf{U}'_E .

Definition 3. *The equivalent GEM for an SVGEM includes:*

- A directed (possibly cyclic) graph with:
 - A node E_i for every event label $E_i \in \mathcal{E}$ and a node $E_j^{s,s'}$ for every possible state transition s, s' of every state variable $X_j \in \mathcal{X}$ in the SVGEM
 - Directed arcs as follows: (i) for all E_i , retain arcs in $\mathbf{U}_{E_i}^\mathcal{E}$ and add arcs emanating from all $E_j^{s,s'}$ corresponding to $X_j \in \mathbf{U}_{E_i}^\mathcal{X}$; (ii) for all $E_j^{s,s'}$ corresponding to X_j , add arcs from all $\mathbf{U}_{X_j}^\mathcal{E}$, all $E_k^{s,s'}$ corresponding to $X_k \in \mathbf{U}_{X_j}^\mathcal{X}$, and all $E_j^{s,s'}$ corresponding to X_j (including the self loop).
- Conditional intensity rate parameters as follows:
 - For all E_i , the intensity rate remains the same and is determined by historical arrivals of all parent event labels in the GEM, $\lambda_{E_i|h\mathbf{U}'_{E_i}}(t) = \lambda_{E_i|h\mathbf{U}_{E_i}}(t)$
 - For all $E_j^{s,s'}$ corresponding to X_j , historical dependence is restricted by the set of nodes $\{E_j^{s^0,s''}\}$ where $s'' \neq s$, otherwise it is equal to the corresponding rate in the SVGEM, $\lambda_{E_j^{s,s'}|h\mathbf{U}'_{E_j^{s,s'}}}(t) = \lambda_{X_j|h\mathbf{U}_{X_j}}(t)$

Theorem 4. *The dynamic process represented by an SVGEM is identical to that of its equivalent GEM.*

Proof. (Outline) The proof follows from the construction of the eqv. GEM. Event arrivals in both the SVGEM and the eqv. GEM follow the same process since the label set and corresponding intensity rates are identical. For state variables, there is a new set of event labels in the eqv. GEM, rates for which are restricted based on the constraint around state transitions. This ensures that transition events in the eqv. GEM process occur at the same rate as the SVGEM process only when it is feasible for the transition to happen. \square

An equivalent GEM for an SVGEM replaces state variables with an event label for every possible state variable transition. Def. 3 formalizes the graph construction and determination of the conditional intensity rates in the equivalent GEM. Fig. 1(b) shows the equivalent GEM for the SVGEM in (a). Since it is necessary to add several arcs to correctly capture the temporal dynamics, we avoid diagram clutter by keeping these arcs implicit using boxes over the new event labels.

We argue that an SVGEM is typically a better representation than its equivalent GEM for communicating with the user (or decision maker) because:

1. An SVGEM makes the semantic difference between the two types of variables self-evident.
2. An SVGEM can be much more compact, particularly if there are several possible state variable transitions – with a reduction from $\sum_i (|\text{Val}(X_i)| * |\text{Val}(X_i) - 1|)$ to $|\mathcal{X}|$ additional nodes in the graph – and therefore easier to interpret (see Fig. 1).
3. Incorporating the constraint regarding state variables is trickier and more difficult to identify in the equivalent GEM as compared to the SVGEM (see Def. 3).

The notion of the equivalent GEM is still however a useful tool for computational reasons. For instance, this equivalence can be used to compute the log likelihood while learning specific parametric SVGEMs using temporal datasets.

2.5 Generality of SVGEMs

The following theorem formalizes the generality of the SVGEM framework by specifying some important model families in the literature.

Theorem 5. (i) A GEM [Didelez, 2008; Gunawardana and Meek, 2016] is an SVGEM without state variable nodes. (ii)

A CTBN [Nodelman et al., 2002] is an SVGEM without event label nodes and where state variables are Markov w.r.t their (state variable) parents. (iii) An ECTBN [Bhattacharjya et al., 2020] is an SVGEM with no arcs from state variables to event labels and where state variables are Markov w.r.t their state variable parents.

Proof. (Outline) An SVGEM without state variables only has nodes E with intensity rates $\lambda_{E|h_{\cup E}}(t)$; this is how GEMs are defined. Removing event labels entirely instead and assuming Markov historical dependence results in a conditional Markov process. Incorporating the labels back but disabling arcs from state variables to events results in a set of conditional Markov processes that depend on historical event occurrences; thus the CTBN and ECTBN models are covered by the SVGEM framework as specified. \square

The following result makes a connection between Markov historical dependence in CTBNs (as well as ECTBNs) and piece-wise constant historical dependence.

Theorem 6. *A CTBN and its extension ECTBN with proximal dependence w.r.t event label parents both involve PC historical dependencies for every node.*

Proof. (Outline) In the specified ECTBN, state variables are Markov w.r.t their state variable parents and proximal w.r.t their event label parents. The transition rates therefore change from a constant to another when either the parent condition is modified due to a boundary window condition including an arrival of a parent label, or a transition of a state variable parent. A CTBN only has state variable parents and therefore only changes parameters at parent transitions. Event labels are proximal w.r.t each other in an ECTBN. \square

2.6 Two Specific SVGEMs

In the previous sub-section, we cast various models from the literature as special cases within the SVGEM framework. A case that is notably missing is where event labels are also affected by state variables, along with other event labels. This is common in numerous domains, for instance, a hospital is likely to be visited by patients with chronic conditions more often, and a manufacturing plant operating in varying modes exhibits different event trajectories.

We introduce two specific parametric models that simultaneously allow for influences in both directions. They are simple but practical models, and as we explain in the next section, easily learned from data. For model naming convention, assumptions about historical dependence for state variables are listed before event labels. Thus, the first ‘MP’ in ‘MP_MP’ signifies that state variable nodes are Markov w.r.t their state variable parents and proximal w.r.t their event label parents, and the second ‘MP’ signifies the same for event label nodes.

SVGEM_MP_MP. In this SVGEM, a node is always Markov w.r.t its state variable parents and proximal w.r.t its event label parents. This is a natural generalization of ECTBN where in addition to state variable dynamics that depend on events in a proximal manner, an event’s arrival rate depends on the current state of its state variable parents.

SVGEM_MP_PP. Here, state variable dynamics stay the same but now an event label node is proximal (not Markov) w.r.t its state variable parents. This is a unique view of dependence on state variables, where the historical occurrence time of a parent state variable transition within a recent window, as well as the nature of the transition, determine an event’s arrival rate. In this fashion, historic state transitions are effectively treated as event labels, like in an equivalent GEM of an SVGEM. We consider the following 4 ways to categorize transitions into what are effectively new labels:

- *all*: all transitions are considered equivalent.
- *each*: each type of transition is a unique label.
- *in*: transitions going into a state are equivalent.
- *out*: transitions going out of a state are equivalent.

We show later through experiments that this particular non-Markovian dependence of event arrivals on historical state transitions can often be a suitable fit for real-world data.

Theorem 7. *SVGEM_MP_MP and SVGEM_MP_PP both involve PC historical dependencies for every node.*

Proof. (Outline) State variables in both models have historical dependence similar to ECTBN, so the relevant part of Thm. 6 applies. In SVGEM_MP_MP, event labels are proximal w.r.t event label parents and Markov w.r.t state variable parents. Intensity rates therefore change from a constant to another when either the parent condition is modified due to a boundary window condition including a parent event arrival, or transition of a state variable parent. In SVGEM_MP_PP, event labels are proximal w.r.t an expanded label set and are piece-wise constant w.r.t the new change points. \square

Fig. 2 (Top) illustrates a piece-wise constant profile for the conditional intensity rate for label E_1 , modeled using SVGEM_MP_MP: it is Markov w.r.t X_1 and proximal w.r.t E_2 . The figure indicates points on the timeline where the conditional intensity rate changes. Later we show that the above result is useful computationally during learning, since a piece-wise constant model simplifies the computation of parameters given the graph and other hyper-parameters.

3 Learning SVGEMs

\mathcal{G} , $\mathbf{P}_{\mathcal{X}}^0$ and $\mathbf{\Lambda}$ for an SVGEM can all be learned from data. Similar to models in the GEMs family, the parents for every node in an SVGEM can be learned separately and then combined to form the overall graph \mathcal{G} . For the graph search procedure, we take a hill climbing score-based approach using the Bayesian information criterion (BIC) score, which is used to measure model fit performance in terms of the log likelihood of the model on a given dataset as well as the model complexity; since this approach is standard in the literature [Nodelman et al., 2003], here we focus primarily on the issue of learning parameters $\mathbf{\Lambda}$ given a graph.

If the parents of each node are known, then one can use the equivalent GEM notion of an SVGEM and ascertain that the log likelihood of a model with parameters $\mathbf{\Lambda} = \{\mathbf{\Lambda}_X\}, \{\mathbf{\Lambda}_E\}$ on an event dataset can be factorized as follows:

$$L(D|\mathbf{\Lambda}_X, \mathbf{\Lambda}_E) = \left[\prod_{X \in \mathcal{X}} L(D_X|\mathbf{\Lambda}_X, D_{U_X}) \right] \left[\prod_{E \in \mathcal{E}} L(D_E|\mathbf{\Lambda}_E, D_{U_E}) \right]$$

Dataset	$ \mathcal{X} $	$ \mathcal{E} $	$\sum_i N_i$	N_E
Synthetic	5	5	$\sim[35-45.8]K$	$\sim[15.1-16.9]K$
Eastman	16	18	$\sim[84-102.3]K$	$\sim[22.6-25.4]K$
Diabetes	1	11	$\sim 6.2K$	$\sim 14K$
Bitcoin	2	6	0	$\sim 66.4K$

Table 1: Dataset information: # of state variables ($|\mathcal{X}|$), event labels ($|\mathcal{E}|$), state transitions ($\sum_i N_i$) and events (N_E). Ranges are provided for the multiple synthetic and Eastman datasets.

The computation of individual node log likelihoods is determined by the model parametrization, which depends on the assumptions about historical dependence. We summarize these below for our two proposed parametric SVGEMs. We use the following notation to represent an instantiation of parent sets: for a general node Z of either type, we introduce vector $\mathbf{u}_Z = \mathbf{u}_Z^{\mathcal{X}} \times \mathbf{u}_Z^{\mathcal{E}}$, where $\mathbf{u}_Z^{\mathcal{X}}$ is a joint instantiation of Z 's state variable parents, i.e. from $\text{Val}(\mathbf{U}_Z^{\mathcal{X}})$, and $\mathbf{u}_Z^{\mathcal{E}}$ is a binary vector of indicators of Z 's event label parents, each depending on whether it is present in some recent window.

SVGEM_MP_MP. The parameters in this model are $\Lambda = \{\lambda_{E|\mathbf{u}_E}, Q_{X|\mathbf{u}_X}\} \forall E, X$, which includes intensity rates for event labels E for each parent instantiation \mathbf{u}_E and a set of conditional matrices for state variables X , one for each parent instantiation \mathbf{u}_X . From Thm. 7, this parametrization results in piece-wise constant historical dependence. One can therefore learn estimates for intensity rates through summary statistics on an event dataset [Gunawardana *et al.*, 2011].

For example, for parent instantiation $u_{E_1} = \{u_{E_1}^{X_1} = \text{low}, u_{E_1}^{E_2} = 0\}$ of label E_1 , the maximum likelihood estimate computed on the event dataset in Figure 2 (Bottom) for $\hat{\lambda}_{E_1|u_{E_1}}$ is $1/2$. This is the rate at which E_1 occurs given that X_1 is in state ‘low’ and that E_2 has not happened within the last 5 days. It is obtained by counting the number of occurrences of E_1 under the condition u_{E_1} and dividing by the duration over the timeline when the condition is true.

SVGEM_MP_PP. In this SVGEM, there is an effective expanded label set $\mathcal{E}' \supset \mathcal{E}$, some of which are parents for event labels. Since event labels are proximal w.r.t parent state variable transitions, which are treated as events in various ways (all/in/out/each), they have a new effective event label parent set U_E^* . Note that the parent set for E in the ‘each’ variation is identical to that in the equivalent GEM for an SVGEM, where each transition is a separate event label.

Suppose \mathbf{u}_E^* is a binary vector of indicators specifying which event label parents have occurred in some recent window. The parameters in this model are $\Lambda = \{\lambda_{E|\mathbf{u}_E^*}, Q_{X|\mathbf{u}_X}\} \forall E, X$. Again from Thm. 7, the historical dependence in this model is piecewise-constant and one can use summary statistics to obtain maximum likelihood estimates of the parameters from an event dataset.

4 Experiments

We test the two parametric SVGEMs by evaluating how well they fit simulated and real-world joint temporal datasets. We consider datasets that have been simulated or processed from

publicly available sources; experimental details around processing and hyper-parameter choices are omitted here due to space restrictions but will be available on the arXiv version.

4.1 Datasets

Table 1 summarizes information about all datasets.

Synthetic. We generate synthetic joint temporal datasets with known graph and parameters from an underlying SVGEM_MP_MP process. We dynamically generate both state variable transitions and event label arrivals based on historical occurrences, with modifications to the procedures outlined in Nodelman *et al.* [2003] and Bhattacharjya *et al.* [2018] respectively. The main adjustment is to also jointly account for the current states of parent state variables and the current conditions of parent event labels at any time, and to have occurrences of both types compete with each other during generation.

For these experiments, we consider 3 datasets, each with 5 state variables and 5 event labels, with different graphs and parameters; the parental set for each node is randomly chosen with a limit of parental size up to 5 for event label parents and 3 for state variable parents. Each dataset includes 10 independent streams generated up to time horizon $T = 5000$.

Tennessee Eastman (TE) Process. TE is a well known process in the control literature, involving two irreversible chemical reactions that produce two liquid products from four gaseous reactants. There are multiple interacting processing units such as a reactor, condenser, compressor, etc., as well as multiple feedback controllers that seek to maintain production rate, quality and safety set points [Downs and Vogel, 1993]. We use a MATLAB simulator¹ of this process to simulate several process and control variables in the presence of numerous distinct process faults and Gaussian process noise.

We generate 2 data sets that each involve a random number of process faults with random start and end times over a horizon of 50 hours, with simulated data generated every 0.6 minutes. We focus attention on 9 controls and 16 process variables, discretizing their simulated trajectories into 3 bins each. Each control variable transition across bins is treated as one of two event types, namely control-up and control-down, depending on whether the control transition is to a higher or lower value. This leads to a system with 16 state variables with 3 states each and 18 event labels (2 for each control variable), mimicking a real-world system with coupled dynamics between process state variables and control events.

Diabetes. We consider a dataset with information pertaining to 70 diabetic patients [Frank and Asuncion, 2010]; events include insulin dosage, eating and exercise related activities, and the blood glucose level is modeled as the sole state variable; data for the latter is formed from discretization of raw measurements into 3 states.

Bitcoin. This dataset from the SNAP library [Leskovec and Krevl, 2014] involves ratings between users on a Bitcoin exchange. We process the data from the perspective of each user, associating them with 6 event labels depending on

¹<http://depts.washington.edu/control/LARRY/TE/download.html>

Dataset	SVGEM-MP-MP	SVGEM-MP-PP	CTBN+PGEM	CTBN+PCIM	PGEM-each
Synth#1	-15846	-16441 (each/in/out)	-16507	-16499	-17985
Synth#2	-19868	-20201 (each)	-20961	-20957	-23602
Synth#3	-15959	-16377 (each/in/out)	-16439	-16447	-17991
Eastman#1	-1420	-1236 (in)	-1432	-1560	-3183
Eastman#2	-836	-684 (out)	-1034	-1352	-1255
Diabetes	-3223	-2456 (all)	-3740	-3556	-3708
Bitcoin	-45298	N/A	-45612	-42321	N/A

Table 2: Log likelihood (LL) for the models on the test sets. For SVGEM_MP_PP, the best performing variations are noted. For Bitcoin, state variables play no role in the LL computation (no transitions), thus the 3^{rd} and 4^{th} columns are really events-only PGEM and PCIM.

whether they sent or received a rating, and whether the rating was positive, neutral or negative, as determined by discretizing ratings between -10 and 10. The method proposed in Kumar *et al.* [2016] is used to measure the ‘Goodness’ and ‘Fairness’ for each user; these are the state variables, assumed to be constant for each user throughout the time horizon. Note that state variables never transition in this dataset.

4.2 Baselines

As far as we are aware, this is the first work to propose a joint bi-directional graphical modeling framework for temporal datasets including state variable transitions and event arrivals. We consider the following baselines:

- **CTBN + PGEM:** In this baseline, state variables and event labels are modeled independently; specifically, state variable transitions are learned using a CTBN [Nodelman *et al.*, 2003] and event label arrivals are learned using a PGEM [Bhattacharjya *et al.*, 2018].
- **CTBN + PCIM:** This is as above, but here event label arrivals are modeled using a piece-wise constant intensity model learner [Parikh *et al.*, 2012].
- **PGEM-each:** In this model, every state transition in the dataset is modeled as separate types of events, and then the PGEM learner is run to fit the new dataset. Note that this is slightly different from the equivalent GEM (whose process is equivalent to the corresponding SVGEM) since there is no explicit accounting for any constraints between events.

We also considered a model involving an additional state variable with as many states as event labels, where transitions occur at event arrival epochs. However, since the additional variable would contain all event labels, this model cannot explicitly determine the subset of event labels that affect another, defeating the entire purpose of the SVGEM framework; hence, this was not considered to be a suitable baseline.

4.3 Results

We split each dataset three-ways by event stream into train (70%), dev (15%) and test (15%) sets, optimize each model’s hyper-parameters from a grid using the train/dev sets, and then learn the final model on the train set. A model’s performance is evaluated based on how well it fits the held-out test set using log likelihood.

Table 2 shows the performance of the proposed models in comparison with the baselines across all data sets:

- The SVGEM_MP_MP is indeed the best performer for the synthetic datasets, as expected, since the data was generated from the associated underlying process.
- TE process data entails sequential dependency between controls and the ensuing dynamic response in the process state variables, which in turn excites the feedback controllers for subsequent control actions. The proposed SVGEM framework is well suited for unveiling such co-evolving causal dynamics. In particular, the ‘in’ and ‘out’ variations of the SVGEM_MP_PP model fit the coupled process best since control events occur due to recent changes in the state of the process state variables.
- The ability of SVGEM_MP_PP to capture physical processes is amplified by its performance on Diabetes.
- The Bitcoin dataset is special in that it involves no transitions; as a result, some models are not applicable (shown as N/A) and log likelihood is computed only for the event labels. State variables simply serve here to further distinguish various event label conditional intensities in SVGEM_MP_MP. We observe that while there are benefits of incorporating state variables on top of the events-only PGEM model (compare columns one and three), these are overtaken in this dataset by the more general PCIM representation. We note however that the SVGEM framework is general and allows for embedding a PCIM to capture dependencies b/w event labels. We expect that such a state-variable augmented PCIM may prove to be the best performer for certain data sets.

5 Conclusions

We have introduced a general unifying framework that expands the family of graphical event models to allow for state variable effects, enabling models where they co-evolve together with event arrivals. We have theoretically shown how the proposed framework incorporates many existing models for continuous-time processes and provides a more compact representation with fewer parameters. A representation that learns the dynamic relationships between event labels and state variables can be useful in numerous applications, notably those that involve variables exhibiting stochastic transitions among a set of finite states in the co-existing presence of events; this was demonstrated in the empirical evaluations. Future work in this area may find it beneficial to study models with more complex historical dependencies.

References

- [Bhattacharjya *et al.*, 2018] D. Bhattacharjya, D. Subramanian, and T. Gao. Proximal graphical event models. In *Adv. Neur. Inf. Process. Syst. (NeurIPS)*, pages 8147–8156, 2018.
- [Bhattacharjya *et al.*, 2019] D. Bhattacharjya, K. Shanmugam, T. Gao, N. Mattei, and K. R. Varshney. Event-driven continuous time Bayesian networks: An application in modeling progression out of poverty through integrated social services. In *IJCAI Workshop on AI for Social Good (AI4SG)*, 2019.
- [Bhattacharjya *et al.*, 2020] D. Bhattacharjya, K. Shanmugam, T. Gao, N. Mattei, K. R. Varshney, and D. Subramanian. Event-driven continuous time Bayesian networks. In *Proc. of Conf. on Artif. Intell. (AAAI)*, 2020.
- [Cox and Lewis, 1972] D. R. Cox and P. A. W. Lewis. Multivariate point processes. In *Proc. of the Sixth Berkeley Symposium on Math. Stat. and Prob., Vol. 3: Probability Theory*, pages 401–448, 1972.
- [Dean and Kanazawa, 1989] T. Dean and K. Kanazawa. A model for reasoning about persistence and causation. *Computational Intelligence*, 5:142–150, 1989.
- [Didelez, 2008] V. Didelez. Graphical models for marked point processes based on local independence. *J. R. Stat. Soc. B*, 70(1):245–264, 2008.
- [Downs and Vogel, 1993] J. J. Downs and E. F. Vogel. A plant-wide industrial process control problem. *Computers & Chemical Engineering*, 17(3):245–255, 1993.
- [Du *et al.*, 2016] N. Du, H. Dai, R. Trivedi, U. Upadhyay, M. Gomez-Rodriguez, and L. Song. Recurrent marked temporal point processes: Embedding event history to vector. In *Proc. SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD)*, pages 1555–1564, 2016.
- [Eichler, 1999] M. Eichler. *Graphical Models in Time Series Analysis*. PhD thesis, University of Heidelberg, Germany, 1999.
- [Frank and Asuncion, 2010] A. Frank and A. Asuncion. UCI machine learning repository, 2010.
- [Gao *et al.*, 2020] T. Gao, D. Subramanian, K. Shanmugam, D. Bhattacharjya, and N. Mattei. A multi-channel neural graphical event model with negative evidence. In *Proc. of Conf. on Artif. Intell. (AAAI)*, 2020.
- [Goulding *et al.*, 2016] J. Goulding, S. Preston, and G. Smith. Event series prediction via non-homogeneous Poisson process modelling. In *Proc. of IEEE Int. Conf. on Data Mining (ICDM)*, pages 161–170, 2016.
- [Gunawardana and Meek, 2016] A. Gunawardana and C. Meek. Universal models of multivariate temporal point processes. In *Proc. of Int. Conf. on Artif. Intell. Stat. (AISTATS)*, pages 556–563, 2016.
- [Gunawardana *et al.*, 2011] A. Gunawardana, C. Meek, and P. Xu. A model for temporal dependencies in event streams. In *Adv. Neur. Inf. Process. Syst. (NeurIPS)*, pages 1962–1970, 2011.
- [Kolar *et al.*, 2010] M. Kolar, L. Song, A. Ahmed, and E. P. Xing. Estimating time-varying networks. *The Annals of Applied Statistics*, 4(1):94–123, 2010.
- [Kumar *et al.*, 2016] S. Kumar, F. Spezzano, V. S. Subramanian, and C. Faloutsos. Edge weight prediction in weighted signed networks. In *Proc. of Int. Conf. on Data Mining (ICDM)*, pages 221–230. IEEE, 2016.
- [Leskovec and Krevl, 2014] J. Leskovec and A. Krevl. SNAP Datasets: Stanford large network dataset collection. <http://snap.stanford.edu/data>, June 2014.
- [Meek, 2014] C. Meek. Toward learning graphical and causal process models. In *Proc. of UAI Workshop on Causal Inference: Learning and Prediction*, pages 43–48, 2014.
- [Murphy, 2002] K. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis, University of California Berkeley, USA, 2002.
- [Nodelman *et al.*, 2002] U. Nodelman, C. R. Shelton, and D. Koller. Continuous time Bayesian networks. In *Proc. of Conf. on Uncertainty Artif. Intell. (UAI)*, pages 378–378, 2002.
- [Nodelman *et al.*, 2003] U. Nodelman, C. R. Shelton, and D. Koller. Learning continuous time Bayesian networks. In *Proc. of Conf. on Uncertainty Artif. Intell. (UAI)*, pages 451–458, 2003.
- [Parikh *et al.*, 2012] A. P. Parikh, A. Gunawardana, and C. Meek. Conjoint modeling of temporal dependencies in event streams. In *Proc. of UAI Workshop on Bayesian Modeling Applications*, August 2012.
- [Pearl, 2014] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 2014.
- [Plis *et al.*, 2015] S. Plis, D. Danks, C. Freeman, and V. Calhoun. Rate-agnostic (causal) structure learning. In *Adv. Neur. Inf. Process. Syst. (NeurIPS)*, pages 3303–3311, 2015.
- [Rajaram *et al.*, 2005] S. Rajaram, T. Graepel, and R. Herbrich. Poisson-networks: A model for structured point processes. In *Proc. Int. Workshop Artif. Intell. Stat. (AISTATS)*, pages 277–284, 2005.
- [Simma and Jordan, 2010] A. Simma and M. I. Jordan. Modeling events with cascades of Poisson processes. In *Proc. of Conf. on Uncertainty Artif. Intell. (UAI)*, pages 546–555, 2010.
- [Weiss and Page, 2013] J. C. Weiss and D. Page. Forest-based point process for event prediction from electronic health records. In *Machine Learning and Knowledge Discovery in Databases*, pages 547–562, 2013.
- [Xiao *et al.*, 2017] S. Xiao, J. Yan, X. Yang, H. Zha, and S. M. Chu. Modeling the intensity function of point process via recurrent neural networks. In *Proc. of Conf. on Artif. Intell. (AAAI)*, pages 1597–1603, 2017.
- [Zhou *et al.*, 2010] S. Zhou, J. Lafferty, and L. Wasserman. Time varying undirected graphs. *Machine Learning*, 80(2–3):295–319, 2010.