A Comparative Review of Probabilistic Inference for Timescale Graphical Event Models

El Mokhtar Hribach EL-MOKHTAR.HRIBACH@UNIV-NANTES.FR
Nantes Université, CNRS, LS2N UMR 6004, F-44000 Nantes, France

Julien Blanchard JULIEN.BLANCHARD@UNIV-NANTES.FR
Nantes Université, CNRS, LS2N UMR 6004, F-44000 Nantes, France

Philippe Leray PHILIPPE.LERAY@UNIV-NANTES.FR
Nantes Université, CNRS, LS2N UMR 6004, F-44000 Nantes, France

Editors: Gustau Camps-Valls, Manuele Leonelli and Gherardo Varando

Abstract

Timescale Graphical Event Models (TGEMs) provide a continuous-time probabilistic framework for representing dependencies among multivariate point processes using a directed graph, where each node corresponds to an event label and each edge specifies how occurrences of a parent event label influence the conditional intensity of a child event label over future time intervals, called timescales. Although TGEMs have been mainly investigated from a modeling and learning perspective, the practical behavior of sampling-based inference methods in these models remains insufficiently characterized. In this paper, we empirically compare three sampling strategies for probabilistic inference in TGEMs: forward sampling, Poisson Superposition Importance Sampling (PSIS), and auxiliary Gibbs sampling. We evaluate these methods on episodic sequence queries with incomplete and imprecise evidence. We examine the consistency of probability estimates, convergence behavior, and sensitivity to the rarity of the evidence. Our results provide a comparative experimental framework for assessing the strengths and limitations of each strategy and offer practical guidance for method selection in TGEM-based inference.

Keywords: Timescale Graphical Event Models, probabilistic inference, continuous-time event models, sampling-based inference, episodic sequence queries.

1. Introduction

Temporal data arise in many domains as ordered sequences of discrete events indexed by continuous time, e.g. financial transactions, online user interactions, communication network logs, and clinical event histories. Modeling these data is valuable for understanding dependencies among events and forecasting future events. Several probabilistic models have been proposed to represent temporal dependencies in continuous time, including Continuous Time Bayesian Networks (Nodelman et al., 2002), Poisson Networks (Rajaram et al., 2005), Piecewise-Constant Conditional Intensity Models (Gunawardana et al., 2011), and Graphical Event Models (Meek, 2014). These models provide a principled way to describe how the past history of a process influences future event intensities. In particular, Timescale Graphical Event Models TGEMs (Gunawardana and Meek, 2016) specify temporal dependencies through time windows, called timescales, associated with the edges of the graphical model. In this paper, we mainly refer to TGEMs since they subsume earlier event-oriented models

© 2026 E. Hribach, J. Blanchard & P. Leray.

such as Poisson Networks and Piecewise-Constant Conditional Intensity Models (excluding CTBNs, in which the events relate exclusively to state variable transitions).

Research efforts on all these models have primarily focused on learning tasks, such as estimating model parameters, learning the graph structure, or identifying temporal dependencies and relevant timescales. In this paper, we focus on the inference task: given a model and an event sequence (complete evidence) or pieces of an event sequence (incomplete evidence), the objective is to answer queries about unobserved parts of the sequence. For instance, in a clinical monitoring setting, a temporary monitor failure may create an unobserved period, after which only partial information about the patient's subsequent events is available. Estimating a target event probability during the missing period then requires jointly accounting for the observations recorded before and after the failure. What methods could be used in this case? And how effective are they? This paper is a first study aimed at answering these questions.

Exact inference in continuous-time probabilistic models is generally difficult, since it requires marginalizing over event sequences with a variable number of events occurring at continuous times. A key difficulty is the *entanglement* (Nodelman et al., 2002; Rajaram et al., 2005) introduced by temporal dependencies. For example, suppose that an event B, such as "I'm buying an umbrella", depends on the past events A, such as "it is raining", over the last three days, then B is linked to A at every instant during the previous three days. Viewed as a Dynamic Bayesian Network, it means that B is connected to an infinite number of (instantaneous) time slices. This entanglement prevents any marginalization over a hidden variable by integration. This fundamental difficulty motivates the use of approximate inference methods, and sampling-based approaches in particular.

Rather than proposing a new sampling algorithm, this paper provides an empirical comparative study of how existing sampling-based inference methods for continuous-time event models behave. These methods have been introduced for distinct types of models and mostly been studied in separate settings: forecasting from a complete observed history, rare episodic sequence estimation, or reconstruction of partially observed trajectories. To compare the methods, we propose in this paper to employ a two-pronged generalization:

- we use TGEMs in order to encompass the various types of models,
- we use a novel type of query, which we call ES-IIE, capable of handling the different usage conditions inherent to each method.

More precisely, in an ES-IIE query, the target is an Episodic Sequence and the evidence is an Incomplete and Imprecise Evidence that provides information both to the left and to the right of the target (before and after the target). With this type of query, we are able to compare the sampling methods in terms of estimated query probabilities, convergence behavior and computational cost, with a particular focus on their ability to incorporate future information.

The remainder of the paper is organized as follows. Section 2 introduces temporal event sequences, Graphical Event Models, Timescale Graphical Event Models, and the sampling-based inference methods considered in this work. Section 3 presents the ES-IIE query type and its unified inference framework. Section 4 describes the experimental protocol and discusses the result. Section 5 concludes the paper and outlines future research directions.

2. State of the Art

This section introduces the background required for the inference problem studied in this paper. We first define temporal event sequences and conditional intensities, then present Graphical Event Models and Timescale Graphical Event Models, which are the modeling framework used in our experiments. Finally, we discuss related inference tasks and sampling-based methods that are used in our comparison.

We consider data in the form of temporal event sequence. Each event is a pair (t_i, l_i), where $t_i \in R^+$ is its occurrence time, and $l_i \in \mathcal{L}$ denotes its label, drawn from a finite set of event labels $\mathcal{L}$. An event sequence observed up to time t^* is then defined as an ordered set of labeled events:

$$x_{t^*} = \{(t_i, l_i)\}_{i=1}^{|x_{t^*}|}$$

such that: $0 < t_1 < t_2 < \cdots < t_{|x_{t^*}|} < t^*$. The history of the event sequence at time t is defined as the sub-sequence of events that occurred strictly before t:

$$h(t) = \{(t_j, l_j) : t_j < t\}$$

A filtered history is the restriction of this history to a subset of event labels. For any subset $\mathcal{S} \subseteq \mathcal{L}$, it is defined as:

$$[h(t)]_{\mathcal{S}} = \{(t_j, l_j) \in h(t) : l_j \in \mathcal{S}\}.$$

2.1. Graphical Event Models

Following the foundational work on Poisson Networks (Rajaram et al., 2005) and Piecewise-Constant conditional Intensity Models (Gunawardana et al., 2011), Graphical Event Models (GEMs) were introduced by Meek (2014) to extend the principles of Bayesian networks (Pearl, 1988) to continuous-time event data. Like their two precursor models, GEMs assume that the temporal point process associated with each label is governed by a conditional intensity function $\lambda_l(t \mid h(t))$ which is the density (instantaneous occurrence rate) of the label l at time t given the history $h(t)$, i.e. the realization of the process over [0,t). This weak assumption is commonly used since it makes it feasible to express the point process likelihood functions (Daley and Vere-Jones, 2008).

A GEM is defined as a pair $\mathcal{M} = (G, \Lambda)$, where $G = (\mathcal{L}, E)$ is a directed graph over a finite set of event labels $\mathcal{L}$, and $\Lambda = \{\lambda_l\}_{l \in \mathcal{L}}$ is a set of conditional intensity functions. Each node $l \in \mathcal{L}$ corresponds to a stochastic process with intensity $\lambda_l(t \mid h(t))$ describing the occurrence of events of label l, and each directed edge $(l', l) \in E$ indicates that the past occurrences of events of label l' affect the intensity of event label l. The graph describes all the potential direct dependencies among event labels, in order that:

$$\lambda_l(t \mid h(t)) = \lambda_l\big(t \mid [h(t)]_{\mathrm{Pa}(l)}\big)$$

where $\mathrm{Pa}(l)$ is the set of parents of l in G. This is analogous to Bayesian Networks, where the conditional probability of a variable given its predecessors depends only on its parents.

For an event sequence $x_{t^*} = \{(t_i, l_i)\}_{i=1}^{|x_{t^*}|}$ observed up to time t^*, the likelihood under a GEM is:

$$p(x_{t^*} \mid t^*) = \prod_{i=1}^{|x_{t^*}|} \lambda_{l_i}(t_i \mid h_i) \prod_{i=1}^{|x_{t^*}|+1} e^{-\sum_{l \in \mathcal{L}} \int_{t_{i-1}}^{t_i} \lambda_l(\tau \mid h_i) d\tau} \tag{1}$$

where $h_i = h(t_i)$. Here we have used the conventions $t_0 = 0$ (start time of observation) and $t_{|x_{t^*}|+1} = t^*$ (end time of observation).

The GEM framework subsumes several earlier approaches to modeling temporal event data, including Continuous-Time Bayesian Networks, Poisson Networks, and Piecewise-Constant Conditional Intensity Models.

2.2. Timescale Graphical Event Models

Timescale Graphical Event Models (TGEMs) are a particular class of GEMs in which the temporal dependence on the past history is specified through time windows called timescales. More precisely, each edge $e = (l', l)$ is associated with a finite temporal horizon $z_e \in \mathcal{R}_+^*$ which is the duration over which l' influences l. Then a timescale is a set of consecutive half-open intervals $I \subset \mathcal{R}_+^*$ of the form $(a, b]$ that make up a partition of the interval $(0, z_e]$.

A TGEM $\mathcal{M} = (G, \Lambda, T)$ is a GEM (G, Λ) augmented with the set $T = \{T_e\}_{e \in E}$ containing the timescales associated to each edge of G. The intensities $\lambda_l(t \mid h(t))$ are piecewise-constant in time and their (plateau) value at time t depends on the parent events of l which have occurred in the timescale intervals on the left of t:

$$\lambda_l\big(t \mid [h(t)]_{\mathrm{Pa}(l)}\big) = \lambda_{l|c_l(t,h(t))} \tag{2}$$

For a label $l \in \mathcal{L}$, the *parent count vector* $c_l(t, h(t))$ is the vector containing the number of occurrences of a parent label l' inside a timescale interval of the edge (l', l), for all the parents l' of l and all the intervals of the timescale associated with the edge (l', l). In the end, for a TGEM, specifying the set Λ of conditional intensity functions amounts to specifying (for each label l) the set of possible constant values $\lambda_{l|\bullet}$. This set is indexed by the possible configurations[1] of the parent count vector c_l. This is analogous to Bayesian Networks, where the conditional probability parameters are indexed by parent states.

Let C_l denote the set of all possible parent count vectors for event type l. Since the intensity of a label l is constant given the parent count vector $c \in C_l$, the data likelihood now factorizes over labels and parent count vectors. For an event sequence $x_{t^*} = \{(t_i, l_i)\}_{i=1}^{|x_{t^*}|}$ observed up to time t^*, the likelihood under a TGEM is:

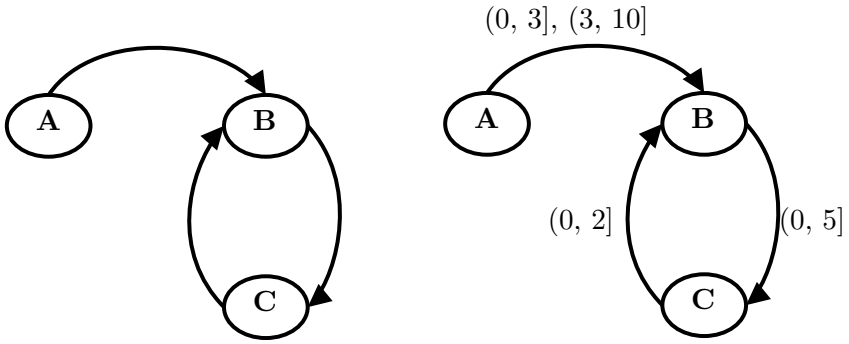
$$p(x_{t^*} \mid t^*) = \prod_{l \in \mathcal{L}} \prod_{c \in C_l} \lambda_{l,c}^{n_{t^*,l,c}} \, e^{-\lambda_{l,c} d_{t^*,l,c}} \tag{3}$$

where:

- $n_{t^*,l,c}$ is the number of events of label l in x_{t^*} that occurred when the parent count vector was in configuration c ;
- $d_{t^*,l,c}$ is the duration up to time t^* during which the parent count vector was in configuration c.

In Figure 1, the edge A to B is associated with two timescale intervals, whereas the edge C to B is associated with one interval. Consequently, the parent count vector of B contains three binary components: two encoding occurrences of A and one encoding occurrences of C in their respective timescale intervals. Therefore, $\mathcal{C}_B = \{0, 1\}^3$, yielding eight possible parent configurations and eight corresponding conditional intensity parameters for event label B.

1. In the parent count vector, counts are thresholded to limit configurations. They are generally binary, indicating either zero occurrence of the parent label or at least one occurrence.



(a) GEM (b) TGEM

Figure 1: An illustrative example of a GEM structure and a TGEM defined on the same graph. (a) The GEM specifies the dependencies between event labels. (b) The TGEM additionally assigns timescales to each edge, specifying the temporal windows over which parent event may affect child intensities. Here, $\mathcal{L} = \{A, B, C\}$, $E = \{(AB), (BC), (CB)\}$, $T_{AB} = \{(0, 3], (3, 10]\}$, $T_{BC} = \{(0, 5])\}$, and $T_{CB} = \{(0, 2]\}$.

2.3. Approximate Inference in TGEMs

In the literature, one can find several types of queries for continuous-time event models. Queries are defined by their target (information looked for) and their evidence (events observed or presumed to have been observed). The target may be an average waiting time between events, or an average number of events, or a probability over events (see Table 1 for examples). Gunawardana et al. (2011) introduced a general type of target called *episodic sequence*, which is a sequence of labels associated with temporal intervals $e = \left\{\left(l_j^*, [a_j^*, b_j^*)\right)\right\}_{j=1}^{k}$. The goal of a query with such a target is to estimate the probability that each label l_j^* occurs (at least once) in its interval $[a_j^*, b_j^*)$.

Regarding the evidence, we distinguish between complete evidence and incomplete evidence. In a complete evidence, the observation period of any label is a time interval. In an incomplete evidence, the observation period may be a union of intervals, i.e. there exists at least one label for which there is a gap in the observation sequence. Note that the evidence may be empty (empty intervals for a few labels, or even for all the labels), but in this case the authors implicitly assume that no event was observed before t_{start} (the time at which the sampling begins), which amounts to declaring an evidence filled with blank space.

Reference	Target	Evidence	Methods applied
(Rajaram et al., 2005)	evolution of λ over time	complete	forward sampling
(Gunawardana et al., 2011)	*episodic sequence* probability	complete	forward sampling, importance sampling
(Qin and Shelton, 2015)	average number of occurrences of a label in a time interval	incomplete	Gibbs sampling

Table 1: Examples of queries for continuous-time event models.

We now review the sampling methods that allow to perform approximate inference in continuous-time event models: forward sampling, importance sampling and Gibbs sampling. Gibbs sampling is used to answer a query with incomplete evidence, unlike the other two, which require a complete evidence.

2.3.1. Forward Sampling

Forward sampling is a procedure used to simulate complete event sequences by generating events successively from left to right in time, according to the conditional intensities specified by the model. With forward sampling, the evidence has to be complete, and the generated events extend the evidence on the right.

For Poisson Networks, Rajaram et al. (2005) introduced an exact sampling mechanism that exploits the piecewise-constant nature of the conditional intensity functions. Since TGEMs also define piecewise-constant intensities, the same generative principle can be applied in this framework.

2.3.2. Importance Sampling

Forecasting rare episodic sequences using standard forward sampling presents significant computational challenges, since only a small proportion of the generated trajectories satisfy the query, resulting in noisy probability estimates. To address this issue, Gunawardana et al. (2011) introduced the Poisson Superposition Importance Sampler (PSIS), an importance sampling method designed for forecasting rare episodic sequences in continuous-time event models. The main idea is to forward-sample trajectories from a proposal distribution that increases the probability of satisfying the episodic sequence query, rather than sampling directly from the original model distribution. Each sampled trajectory is then weighted to correct for the difference between the proposal distribution and the target model. In PSIS, the proposal distribution is obtained by modifying the conditional intensity functions with an additional Poisson component that biases the sampling process toward trajectories satisfying the query.

2.3.3. Gibbs Sampling

Gibbs sampling is a Markov Chain Monte Carlo (MCMC) technique designed to generate samples from conditional distributions when direct inference is difficult. This is the case for our models when the evidence is incomplete, since the left-to-right nature of forward sampling makes it impossible to take the evidence on the right side into account For PCIMs, Qin and Shelton (2015) introduced an auxiliary Gibbs sampler based on the thinning principle for inhomogeneous Poisson processes. The method alternates between (1) sampling new virtual events in the unobserved intervals, and (2) retaining only a subset of virtual events in order to generate a likely sequence. The drawback of this method is that it is computationally expensive, since selecting virtual events requires considering all possible configurations.

3. Unifying Framework: a new Query Processable by all Inference Methods

To compare the sampling-based inference methods presented in the previous section, we need a type of query that can be processed by both algorithms designed for complete evidence and those designed for incomplete evidence. For that reason, we introduce a novel type of query, which we call ES-IIE (Episodic Sequence with Incomplete Imprecise Evidence) with these characteristics:

- the evidence consists of two parts: an observed event sequence E_{left} followed by an imprecise constraint on the events IC_{right}, with an unobserved interval between the two (see Figure 2),
- the target is an episodic sequence e included in the unobserved period between E_{left} and IC_{right},
- the inference problem is the estimation of the following probability:

$$P(e \mid E_{\text{left}}, IC_{\text{right}})$$

Since our goal is to compare the methods with a particular focus on their ability to incorporate future information, we opted for an evidence consisting of two parts. The left-hand part is used by the sampling algorithms to initialize the model before sampling, while the right-hand part conveys future information that is more or less consistent with the model as initialized. The evidence structure we adopted is thus highly general.

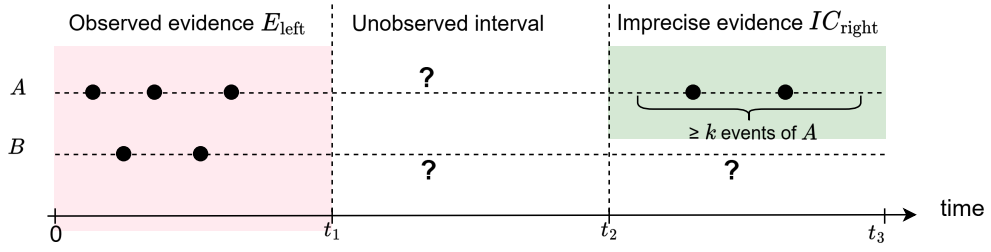


Figure 2: Example of evidence for two labels A and B: a precise evidence E_{left} observed up to time t_1, an unobserved interval, and an imprecise constraint IC_{right} specified over $[t_2, t_3]$.

We refer to the right-hand evidence as *imprecise* because it does not precisely specify the events in the sequence, but provides only a binary constraint that the sampled sequence must satisfy, describing the sequence in a degraded way. In this paper, without loss of generality, we focus on constraints of the form "*at least k events of label A occurred during* $[t_u, t_v]$". Any other constraint in the form of a binary condition on a sequence interval would be possible, such as "*never more than a 2-second wait between two consecutive events during* $[t_u, t_v]$". The probability $P(IC_{\text{right}} \mid E_{\text{left}})$ defines the difficulty of the sampling task conditioned by E_{left}, which is why we use it as an experimental factor in Section 4.

The sampling-based inference methods are able to process an ES-IIE query in this way:

- Forward sampling and PSIS generate trajectories from left to right and therefore cannot directly condition on information located to the right of the generated interval. However, since IC_{right} is imprecise, it can be taken into account by rejection: generated trajectories that do not satisfy IC_{right} are discarded, and the target probability is estimated from the retained trajectories.

- Auxiliary Gibbs sampling, on the other hand, can condition on right-hand information when it is represented as exact observed evidence. Since IC_{right} is imprecise, it must first be randomly instantiated[2] into a compatible precise right-hand evidence. The sampler then generates trajectories satisfying the resulting evidence configuration and estimates the target probability from these samples.

Consequently, all methods can estimate the same probability $P(e \mid E_{\text{left}}, IC_{\text{right}})$ while handling the right-hand constraint differently. The questions that now arise are the following:

- Do the three methods provide consistent estimates of the target probability?
- How does their convergence change as the probability of the imprecise right-hand evidence varies?
- What is the empirical computational cost of each method in terms of execution time?

4. Experiments

We conduct controlled simulation experiments to compare forward sampling with rejection, PSIS with rejection, and auxiliary Gibbs sampling, on ES-IIE queries. The experiments are designed to study how the three estimators behave as the probability of the imprecise right-hand evidence $P(IC_{\text{right}} \mid E_{\text{left}})$ varies, while keeping the underlying model, the target e, and the observed evidence E_{left} constant. This probability measures how restrictive the imprecise right-hand evidence is and provides an interpretable axis for comparing the convergence behaviors and computational cost of the different inference methods.

4.1. Experimental Protocol

We consider the synthetic two-node TGEM illustrated in Figure 3, with event labels A and B connected by reciprocal directed dependencies, each associated with the timescale interval $(0, 1]$. The conditional intensity parameters are given in the figure.

Regarding the query, the target is the episodic sequence $e = \{(B, [5, 7])\}$. The left-hand evidence E_{left} consists of the event history of labels A and B over the interval $[0, 5]$. The observed event times used in the experiments are $E^A_{\text{left}} = \{0.3654\}$ and $E^B_{\text{left}} = \{0.0724, 0.1983, 0.2501, 0.9424, 0.9651, 2.0884, 2.7940, 3.1388, 3.6787, 3.7790, 3.9254, 4.1947, 4.5870, 4.7053\}$. The unobserved intervals are $[5, 6.5]$ for label A and $[5, 8]$ for label B. The imprecise right-hand evidence IC_{right} specifies that at least k events of label A occur in the interval $[6.5, 8]$, where k is varied to control $P(IC_{\text{right}} \mid E_{\text{left}})$.

Since exact inference is not available for the considered queries, we have estimated the true target probability $P(e \mid E_{\text{left}}, IC_{\text{right}})$ as well as the probability of the imprecise right-hand evidence $P(IC_{\text{right}} \mid E_{\text{left}})$ using large independent reference samples generated by forward sampling with rejection.

Figure 4 illustrates the convergence metric used throughout our experiments. For each estimator, we track the running estimate of the query probability as the total number of generated sequences grows. We define a $\pm 1\%$ tolerance band around $P(e \mid E_{\text{left}}, IC_{\text{right}})$ and measure convergence by the *convergence delay*: the last sequence index at which the running

2. In the present study, the event distribution within the interval under E_{left} is estimated using large independent reference samples generated by forward sampling with rejection.

estimate remains inside the tolerance band. For forward sampling and PSIS, the total number of generated sequences counts all generated trajectories, including those rejected because they do not satisfy IC_{right}. For auxiliary Gibbs sampling, one single generated sequence is obtained after a Gibbs iteration over each label. A smaller convergence delay therefore indicates faster and more stable convergence toward the true probability.

For each IC_{right}, each sampling method is repeated over 10 independent runs with different random seeds. The running probability estimates are then averaged pointwise across repetitions, producing one mean convergence curve for each method. The convergence delay reported in the figures is computed from this mean convergence curve. This averaging reduces the influence of seed-specific fluctuations in the displayed curves.

Our TGEM models and the sampling algorithms used in the experiments were implemented in our C++ library, Pilgrim Evential.

4.2. Results and Discussion

Figure 5 shows the mean convergence delay curves for one representative imprecise right-hand evidence configuration with $P(IC_{\text{right}} \mid E_{\text{left}}) = 0.36$.

The first part of the curves illustrates the effect of rejection on left-to-right generation methods. For PSIS, the initial plateau corresponds to a sequence of generated trajectories that do not satisfy IC_{right} and are therefore rejected, during this period, the running estimate remains almost unchanged. A similar phenomenon can also be observed for forward sampling, although over shorter intervals. In contrast, auxiliary Gibbs sampling directly samples conditionally on the instantiated evidence configuration, which explains why its mean estimate stabilizes earlier in this example.

The three estimators converge toward the same probability, but they differ in the number of simulation steps required to remain inside the $\pm 1\%$ band. In this configuration, auxiliary Gibbs sampling reaches the convergence band first, followed by forward sampling with rejection, whereas PSIS with rejection requires a larger number of simulation steps before stabilizing within the band.

Figure 6 compares the convergence behavior of forward sampling with rejection, PSIS with rejection, and auxiliary Gibbs sampling across several imprecise right-hand evidence configurations. The results are given as a function of the imprecise right-hand evidence probability $P(IC_{\text{right}} \mid E_{\text{left}})$. They show that the convergence behavior of the methods is strongly affected by the probability of the imprecise right-hand evidence. For frequent constraints, forward sampling with rejection reaches the convergence band quickly. However, as $P(IC_{\text{right}} \mid E_{\text{left}})$ decreases, its convergence delay increases sharply, reflecting the growing

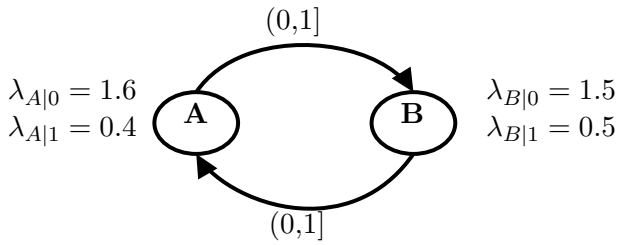


Figure 3: TGEM used in the experiments. The parameters displayed next to each node are the conditional intensity values used in the experiments.

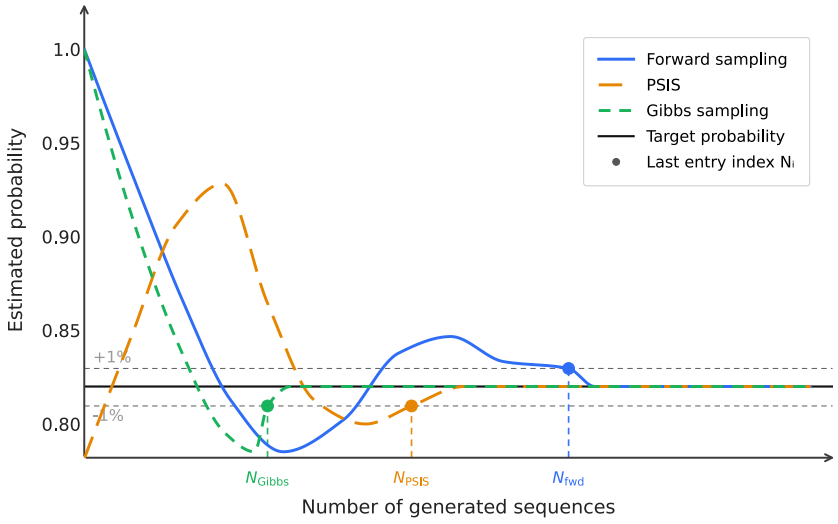


Figure 4: Illustration of the convergence criterion used to compare the sampling methods. The horizontal solid line represents the target probability, the two horizontal dashed lines delimit the $\pm 1\%$ tolerance band around this probability, and each vertical line indicates the convergence delay of an estimator.

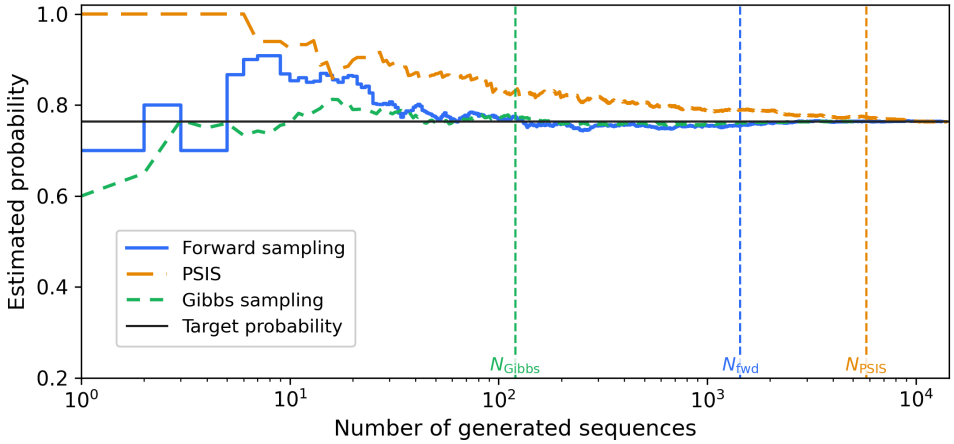


Figure 5: Mean convergence curves over 10 independent repetitions for forward sampling with rejection, auxiliary Gibbs sampling, and PSIS with rejection, with $P(IC_{\text{right}} \mid E_{\text{left}}) = 0.36$. The solid horizontal line represents the target probability and the dashed vertical lines indicate the convergence delay of each estimator.

number of generated trajectories required under rejection. PSIS with rejection also exhibits high convergence delays for rare constraints in this experiment. In contrast, auxiliary Gibbs sampling remains substantially less sensitive to the decrease of $P(IC_{\text{right}} \mid E_{\text{left}})$, with lower convergence delays for the rare constraint configurations. This suggests that conditional resampling over incomplete evidence can be more stable than rejection-based generation when the conditioning event becomes rare.

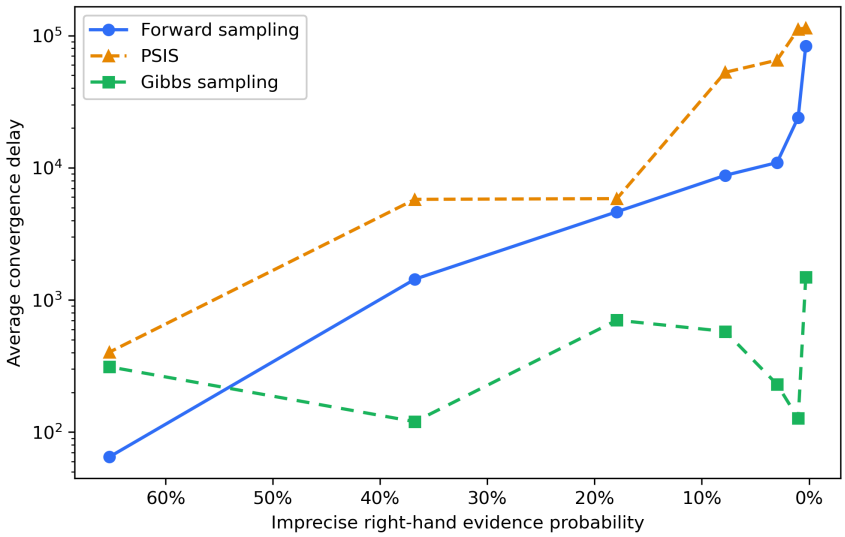


Figure 6: Convergence delay computed from mean convergence curves over 10 independent repetitions for forward sampling with rejection, PSIS with rejection, and auxiliary Gibbs sampling, as a function of the imprecise right-hand evidence probability $P(IC_{\text{right}} \mid E_{\text{left}})$.

It is important to note that the convergence delay reported in Figure 6 is measured in terms of number of generated sequences. To complement this convergence analysis, we measured the average runtime per generated sequence for the three methods across the various imprecise right-hand evidence configurations (Table 2). The runtime experiments were conducted on a compute node equipped with two AMD EPYC 9474F 48-core processors. All runtimes were measured using 16 OpenMP threads pinned to a single socket. In the fourth column of Table 2, the reported Gibbs runtime corresponds to the naive forward-filtering-backward-sampling method of Qin and Shelton (2015), which enumerates candidate trajectories, and not to the dynamic-programming version based on a state-vector representation of the trajectories Qin and Shelton (2015). In the first version, the number of candidate trajectories can grow as $O(2^m)$, whereas state merging reduces this growth to linear in m in most cases in the second version. Profiling confirms that the combinatorial enumeration accounts for approximately 95% of the total Gibbs runtime. To illustrate the potential impact of this optimization, we additionally report $\log_2(t_{\text{Gibbs}})/t_{\text{forward}}$ as a heuristic complexity-based projection in the last column of Table 2. The numerator should not be interpreted as the measured runtime of an optimized implementation, but as an indication of the expected effect of replacing exponential enumeration by near-linear state propagation (a direct empirical evaluation of an optimized implementation is left for future work). Results in Table 2 show the higher computational cost of Gibbs sampling per generated sequence.

These results answer the three questions raised in Section 3. Firstly, the three methods provide consistent estimates of the target probability across all tested configurations. Secondly, their convergence behavior diverges significantly as $P(IC_{\text{right}} \mid E_{\text{left}})$ decreases: rejection-based methods degrade sharply, while auxiliary Gibbs sampling remains less sensitive to the probability of the imprecise right-hand evidence, making it a promising alternative to rejection-based generation when the conditioning event is rare. Thirdly, regarding

$P(IC_{\text{right}} \mid E_{\text{left}})$	forward (μs)	PSIS (multiplier)	naive Gibbs (multiplier)	optimized Gibbs projection (multiplier)
0.65	0.616	×2.27	×10159	×20.47
0.36	0.708	×1.74	×4051	×16.22
0.18	0.578	×1.94	×2943	×18.57
0.078	0.575	×1.88	×2450	×18.19
0.030	0.556	×2.00	×1989	×18.19
0.010	0.551	×1.98	×1750	×17.99
0.003	0.569	×2.02	×1643	×17.34

Table 2: Runtime (per generated sequence) comparison using forward sampling as the reference. Average forward runtimes are reported in μs; PSIS, naive Gibbs and optimized Gibbs are reported as average-runtime multipliers relative to forward sampling. The last column is based on a heuristic complexity-based projection rather than on measured runtimes.

computational cost per generated sequence, forward sampling and PSIS are more efficient than naive auxiliary Gibbs sampling, although this may be counterbalanced by the higher number of iterations highlighted in Figures 5 and 6. Ultimately, these results motivate further investigation of hybrid strategies leveraging the strengths of rejection-based generation and conditional resampling depending on the context.

5. Conclusion

This paper proposed sampling-based inference in TGEMs for episodic sequence queries under a unified evidence formulation combining precise left-hand observations and imprecise right-hand constraint. We empirically compared three methods: forward sampling with rejection, PSIS with rejection, and auxiliary Gibbs sampling. The comparison was conducted across several configurations, obtained by varying the probability of the imprecise right-hand evidence.

Our experiments show that the three methods produce consistent probability estimates across the tested configurations. However, their convergence behavior differs substantially as the imprecise right-hand evidence becomes rare. Forward sampling with rejection and PSIS with rejection require increasingly large number of generated sequences, whereas auxiliary Gibbs sampling is less sensitive to the right-hand evidence probability. Regarding computational cost per generated sequence, forward sampling with rejection and PSIS with rejection are more efficient than auxiliary Gibbs sampling. These results suggest that the choice of inference method should be guided by the probability of the right-hand evidence under the model. When this probability is moderate to high, forward sampling with rejection provides a simple and effective baseline. When this probability is low, auxiliary Gibbs sampling becomes a promising alternative, despite the additional step required to instantiate the imprecise right-hand evidence as precise observed evidence.

Future work could extend this comparison to larger TGEM structures, real-world event datasets, and other evidence schemes. Another promising direction is to investigate other types of queries beyond episodic sequence queries and evaluate how the different sampling methods behave in these settings. Finally, hybrid inference strategies combining the strengths of multiple sampling methods could be explored. Such strategies may provide a better trade-off between computational cost and estimation accuracy.

References

Daryl J Daley and David Vere-Jones. *An introduction to the theory of point processes: volume II: general theory and structure*. Springer, 2008.

Asela Gunawardana and Christopher Meek. Universal models of multivariate temporal point processes. In Arthur Gretton and Christian C. Robert, editors, *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pages 556–563. PMLR, 2016.

Asela Gunawardana, Christopher Meek, and Puyang Xu. A Model for Temporal Dependencies in Event Streams. In *Advances in Neural Information Processing Systems*, volume 24, pages 1962–1970. Curran Associates, Inc., 2011.

Christopher Meek. Toward learning graphical and causal process models. In *Proceedings of the UAI 2014 Conference on Causal Inference: Learning and Prediction - Volume 1274*, CI'14, pages 43—48, Aachen, DEU, 2014. CEUR-WS.org.

Uri Nodelman, Christian R. Shelton, and Daphne Koller. Continuous time Bayesian networks. In *Proceedings of the Eighteenth Conference on Uncertainty in Artificial Intelligence*, UAI'02, pages 378–387, San Francisco, CA, USA, 2002. Morgan Kaufmann Publishers Inc.

Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Mateo, CA, 1988.

Zhen Qin and Christian R. Shelton. Auxiliary Gibbs sampling for inference in piecewise-constant conditional intensity models. In *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence*, UAI'15, pages 722–731, Arlington, Virginia, USA, July 2015. AUAI Press.

Shyamsundar Rajaram, Thore Graepel, and Ralf Herbrich. Poisson-networks: A model for structured point processes. In *Proceedings of the 10th International Workshop on Artificial Intelligence and Statistics*, pages 277–284, 2005.