



TECHNISCHE  
UNIVERSITÄT  
DARMSTADT

Fachbereich Elektrotechnik und Informationstechnik  
Bioinspired Communication Systems

# **Bayesian Inference of Information Transfer in Graph-Based Continuous-Time Multi-Agent Systems**

**Master- Thesis**

Elektro- und Informationstechnik

Eingereicht von

Gizem Ekinici

am

07.07.2020

1. Gutachten: Prof. Dr. techn. Heinz Koeppel
2. Gutachten: Dominik Linzner



## **Erklärung zur Abschlussarbeit gemäß §22 Abs. 7 und §23 Abs. 7 APB TU Darmstadt**

Hiermit versichere ich, Gizem Ekinici, die vorliegende Arbeit gemäß §22 Abs. 7 APB der TU Darmstadt ohne Hilfe Dritter und nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die Quellen entnommen wurden, sind als solche kenntlich gemacht worden. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen. Mir ist bekannt, dass im Falle eines Plagiats (§38 Abs.2 APB) ein Täuschungsversuch vorliegt, der dazu führt, dass die Arbeit mit 5,0 bewertet und damit ein Prüfungsversuch verbraucht wird. Abschlussarbeiten dürfen nur einmal wiederholt werden. Bei der abgegebenen Arbeit stimmen die schriftliche und die zur Archivierung eingereichte elektronische Fassung gemäß §23 Abs. 7 APB überein.

English translation for information purposes only:

Thesis statement pursuant to §22 paragraph 7 and §23 paragraph 7 of APB TU Darmstadt: I herewith formally declare that I, Gizem Ekinici, have written the submitted thesis independently pursuant to §22 paragraph 7 of APB TU Darmstadt. I did not use any outside support except for the quoted literature and other sources mentioned in the paper. I clearly marked and separately listed all of the literature and all of the other sources which I employed when producing this academic work, either literally or in content. This thesis has not been handed in or published before in the same or similar form. I am aware, that in case of an attempt at deception based on plagiarism (§38 Abs. 2 APB), the thesis would be graded with 5,0 and counted as one failed examination attempt. The thesis may only be repeated once. In the submitted thesis the written copies and the electronic version for archiving are pursuant to § 23 paragraph 7 of APB identical in content.

Darmstadt, den 07.07.2020

---

(Gizem Ekinici)



## Abstract

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

# Contents

<b>List of Symbols</b>	<b>i</b>
<b>List of Figures</b>	<b>ii</b>
<b>1. Introduction</b>	<b>1</b>
1.1. Motivation . . . . .	1
1.2. Related Work . . . . .	1
1.3. Contributions . . . . .	1
1.4. Structure of the Thesis . . . . .	1
<b>2. Foundations</b>	<b>2</b>
2.1. Problem Formulation . . . . .	2
2.2. Continuous-Time Bayesian Networks . . . . .	3
2.2.1. Continuous-Time Markov Processes . . . . .	3
2.2.1.1. Homogenous Continuous-Time Markov Processes . . . . .	3
2.2.1.2. Conditional Markov Processes . . . . .	5
2.2.2. The CTBN Model . . . . .	6
2.3. Belief State in Partially Observable Markov Decision Processes . . . . .	6
2.3.1. Exact/Bayes(?) Belief State Update . . . . .	7
2.3.2. Filtering for CTMP . . . . .	7
2.3.3. Belief State Update using Particle Filter . . . . .	8
2.3.3.1. Particle Filtering . . . . .	8
2.3.3.2. Marginalized Continuous-Time Markov Processes . . . . .	9
2.4. Sampling Algorithms . . . . .	10
2.4.1. Gillespie Algorithm for Generative CTBN . . . . .	10
2.4.2. Thinning Algorithm . . . . .	10
<b>3. Experimental Setup</b>	<b>12</b>
3.1. The Model . . . . .	12
3.1.1. CTBN Model . . . . .	12
3.1.2. POMDP Model . . . . .	13
3.1.2.1. Observation Model . . . . .	14
3.1.2.2. Belief State . . . . .	14
3.1.2.3. Optimal Policy . . . . .	15
3.2. Inference of Observation Model . . . . .	16
3.2.1. Likelihood Model . . . . .	16
3.3. Data Generation . . . . .	17
3.3.1. Sampling Algorithm . . . . .	17
3.4. Configurations . . . . .	18

<b>4. Results</b>	<b>20</b>
4.1. Simulation . . . . .	20
4.2. Results . . . . .	21
4.2.1. Maximum Likelihood Estimation . . . . .	21
4.2.2. Maximum Likelihood Classification . . . . .	22
<b>5. Conclusion</b>	<b>24</b>
5.1. Discussion . . . . .	24
5.2. Future Work . . . . .	24
<b>Bibliography</b>	<b>25</b>
<b>A. Amalgamation</b>	<b>26</b>
<b>B. Marginalized Likelihood Function for Homogenous Continuous Time Markov Processes</b>	<b>27</b>

# List of Symbols

$\chi$	state space of random variable $X$
$X(t)$	value of random variable $X$ at time $t$
$X^{[0,T]}$	discrete valued trajectory of random variable $X$ in time interval $[0, T]$
$\mathbf{X}^T$	transpose of matrix/vector $\mathbf{X}$



# List of Figures

2.1. Communication model. . . . .	2
3.1. Hierarchical model. . . . .	12
3.2. Closer look to agent-environment interaction from the perspective of POMDP framework. . . . .	13
4.1. Sampled trajectories . . . . .	20
4.2. Belief state and $Q_3$ trajectories . . . . .	20
4.3. plots of.... . . . .	21
4.4. plots of.... . . . .	21
4.5. plots of.... . . . .	22
4.6. plots of.... . . . .	22
4.7. plots of.... . . . .	23
4.8. plot of... . . . .	23

# **1. Introduction**

## **1.1. Motivation**

## **1.2. Related Work**

## **1.3. Contributions**

## **1.4. Structure of the Thesis**

## 2. Foundations

This chapter presents the theory applied in this thesis. First, the details of the communication problem is described briefly to put the theory into perspective, and then the mathematical theory of the frameworks used to model this problem is introduced.

### 2.1. Problem Formulation

The communication model considered in this thesis is given in Figure ???. The parent nodes,  $X_1$  and  $X_2$ , emit messages which carry information about their states. These messages are translated by an observation model,  $\psi$ , and agent node,  $X_3$  makes a decision based on this translated message,  $y$ . The main objective is to infer the observation model, given a set of trajectories of nodes.

The transition models of the nodes and the dependencies between them are modelled as continuous-time Bayesian network (CTBN), denoted by  $\mathbf{X}$ . The network  $\mathbf{X}$  represents a stochastic process over a structured multivariate state space  $\mathcal{X} = [\chi_1, \dots, \chi_n]$ .

The messages that are emitted by the parent nodes  $X_1$  and  $X_2$  are modelled as independent homogeneous continuous-time Markov processes  $X_i(t)$ , with state space  $\chi_i = \{x_1, x_2, \dots, x_m\}$  for  $i \in \{1, 2\}$ .

The agent node  $X_3$  does not have direct access to the messages but observes a translation of them. The observation model is defined as the likelihood of a translation given the parent messages.

$$\psi := p(y(t) \mid X_1(t), X_2(t)) \quad (2.1)$$

The agent  $X_3$  is modelled as inhomogeneous continuous-time Markov process with state space  $\chi_3 = \{x_1, x_2, \dots, x_m\}$  and set of actions  $a \in \{a_0, a_1, \dots, a_k\}$  to choose from.

Given the observation, the agent forms a belief over the parent states,  $b(x_1, x_2; t)$ , that summarizes the past observations. The policy of the agent,  $\pi(a \mid b)$ , is assumed to be shaped by evolution (close) to optimality. Based on the belief state, the agent takes an action, which in the setting described above means to change its internal dynamics.

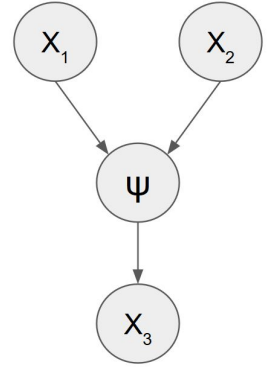


Figure 2.1.: Communication model.

## 2.2. Continuous-Time Bayesian Networks

A continuous-time Bayesian network (CTBN) is a graphical model that represents a collection of nodes whose values evolve continuously over time. In the CTBN framework, through a directed graph, the dependencies of a set of Markov processes (MPs) can be modelled efficiently, relying on two assumptions. The first assumption is that only one node can transition at a time. Secondly, the instantaneous dynamics of each node depends only on its parent nodes. [1, 2]

### 2.2.1. Continuous-Time Markov Processes

A continuous-time Markov process (CTMP) is a continuous-time stochastic process which satisfies Markov property, namely, the probability distribution over the states at a later time is conditionally independent of the past states given the current state.[1] Let  $X$  be a CTMP with state space  $\mathcal{X} = \{x_1, x_2, \dots, x_n\}$ . Then the Markov property can be written as follows:

$$\Pr(X^{(t_k)} = x_{t_k} | X^{(t_{k-1})} = x_{t_{k-1}}, \dots, X^{(t_0)} = x_{t_0}) = \Pr(X^{(t_k)} = x_{t_k} | X^{(t_{k-1})} = x_{t_{k-1}}) \quad (2.2)$$

A CTMP is represented by its transition intensity matrix,  $\mathbf{Q}$ . In this matrix, the intensity  $q_i$  represents the instantaneous probability of leaving state  $x_i$  and  $q_{i,j}$  represents the instantaneous probability of switching from state  $x_i$  to  $x_j$ .

$$\mathbf{Q} = \begin{bmatrix} -q_1 & q_{1,2} & \dots & q_{1,n} \\ q_{2,1} & -q_2 & \dots & q_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ q_{n,1} & q_{n,2} & \dots & -q_n \end{bmatrix} \quad (2.3)$$

where  $q_i = \sum_{j \neq i} q_{i,j}$ . [2]

#### 2.2.1.1. Homogenous Continuous-Time Markov Processes

A continuous-time Markov process is time-homogenous when the transition intensities do not depend on time. Let  $X$  be a homogenous CTMP, with transition intensity matrix  $\mathbf{Q}_X$ . Infinitesimal transition probability from state  $x_i$  to  $x_j$  in terms of the transition intensities  $q_{i,j}$  can be written as [1]:

$$p_{i,j}(h) = \delta_{ij} + q_{i,j}h + o(h) \quad (2.4)$$

where  $p_{i,j}(h) \equiv \Pr(X(t+h) = x_j | X(t) = x_i)$  are Markov transition functions,  $\delta_{i,j} = \delta(x_i, x_j)$  is Kronecker delta and  $o(\cdot)$  is a function decaying to zero faster than its argument.

The *master equation* is then derived as follows:

$$\begin{aligned}
p_j(t) &= \Pr(X(t) = x_j) \\
&= \sum_{\forall i} p_{i,j}(h) p_i(t-h) \\
\lim_{h \rightarrow 0} p_j(t) &= \lim_{h \rightarrow 0} \sum_{\forall i} [\delta_{ij} + q_{i,j}h + o(h)] p_i(t-h) \\
&= \lim_{h \rightarrow 0} p_j(t-h) + \lim_{h \rightarrow 0} h \sum_{\forall i} q_{i,j} p_i(t-h) \\
\lim_{h \rightarrow 0} \frac{p_j(t) - p_j(t-h)}{h} &= \lim_{h \rightarrow 0} \sum_{\forall i} q_{i,j} p_i(t-h) \\
\frac{d}{dt} p_j(t) &= \sum_{\forall i} q_{i,j} p_i(t)
\end{aligned} \tag{2.5}$$

Equation 2.5 can be written in matrix form:

$$\frac{d}{dt} p(t) = p(t) \mathbf{Q} \tag{2.6}$$

where the time-dependent probability distribution  $p(t)$  is a row vector with entries  $\{p_i(t)\}_{x_i \in \mathcal{X}}$ .

**QUESTION:**  $\{\}$  notation for row vectors are used in Prof.Koepl's supporting info paper. I followed it here, but should it be  $[p_i(t)]_{x_i \in \mathcal{X}}$ ?

The solution of this ODE is,

$$p(t) = p(0) \exp(t\mathbf{Q}) \tag{2.7}$$

with initial distribution  $p(0)$ .

The amount of time staying in a state  $x_i$  is exponentially distributed with parameter  $q_i$ . The probability density function  $f$  and cumulative distribution function  $F$  for staying in the state  $x_i$  [2]:

$$f(t) = q_i \exp(-q_i t), t \geq 0 \tag{2.8}$$

$$F(t) = 1 - \exp(-q_i t), t \geq 0 \tag{2.9}$$

Given the transitioning from state  $x_i$ , the probability of landing on state  $x_j$  is  $q_{i,j}/q_i$ .

**Likelihood Function** Consider a single transition denoted as  $d = \langle x_i, x_j, t \rangle$ , where the transition occurs from state  $x_i$  to  $x_j$  after spending  $t$  amount of time at state  $x_i$ . The likelihood of this transition is the product of the probability of having remained at state  $x_i$  for that long, and the probability of transitioning to  $x_j$ .

$$\Pr(d \mid \mathbf{Q}) = (q_i \exp(-q_i t)) \left( \frac{q_{i,j}}{q_i} \right) \tag{2.10}$$

The likelihood of a trajectory sampled from a homogenous CTMC, denoted by  $X^{[0,T]}$ , can be decomposed as the product of the likelihood of single transitions. The sufficient statistics summarizing this trajectory can be written as  $T[x_i]$ , the total amount of time spent in state

$x_i$ ,  $M[x_i, x_j]$  the total number of transitions from state  $x_i$  to  $x_j$ . Then the likelihood of a trajectory  $X^{[0,T]}$  can be written as:

$$\begin{aligned} \Pr(X^{[0,T]} \mid \mathbf{Q}) &= \prod_{d \in X^{[0,T]}} \Pr(d \mid \mathbf{Q}) \\ &= \left( \prod_i q_i^{M[x_i]} \exp(-q_i T[x_i]) \right) \left( \prod_i \prod_{j \neq i} \left( \frac{q_{i,j}}{q_i} \right)^{M[x_i, x_j]} \right) \\ &= \prod_{j \neq i} \exp(-q_{i,j} T[x_i]) q_{i,j}^{M[x_i, x_j]} \end{aligned} \quad (2.11)$$

where  $M[x_i] = \sum_{j \neq i} M[x_i, x_j]$  is the total number transitions leaving state  $x_i$ .

### 2.2.1.2. Conditional Markov Processes

A continuous-time Markov process is *time-inhomogenous* when the transition intensities change over time. In a CTBN, while every node is a Markov process, the leaf nodes are characterized as *conditional* Markov processes, a type of inhomogeneous MP, where the intensities change over time, but not as a function of time rather as a function of parent states. [2]

Let  $X$  be a conditional Markov process, with a set of parents  $\mathbf{U} = \text{Par}(X)$ . Its intensity matrix, *conditional intensity matrix*,  $\mathbf{Q}_{X \mid \mathbf{U}}$  can be viewed as a set of homogenous intensity matrices  $\mathbf{Q}_{X \mid \mathbf{u}}$ , with entries  $q_{i,j \mid \mathbf{u}}$  (similar to Equation 2.3), for each instantiation of parent nodes  $\mathbf{U}(t) = \mathbf{u}$ . [2] As a result, given a trajectory of parent nodes,  $X$  has a trajectory of intensity matrix as

$$\mathbf{Q}^{[0,T]} = [\mathbf{Q}_{X \mid \mathbf{U}(t_0)}, \mathbf{Q}_{X \mid \mathbf{U}(t_1)}, \dots, \mathbf{Q}_{X \mid \mathbf{U}(t_N)}], \quad 0 < t_0 < t_1 < \dots < t_N \leq T. \quad (2.12)$$

Markov transition function for a conditional Markov process can be written as follows:

$$\Pr(X(t+h) = x_j \mid X(t) = x_i, \mathbf{U}(t) = \mathbf{u}, \mathbf{Q}_{X \mid \mathbf{u}}) = \delta(i, j) + q_{i,j \mid \mathbf{u}} h + o(h) \quad (2.13)$$

**Likelihood Function** Given the instantiation of its parents, the complete information on the dynamics of  $X$  is obtained. Then the likelihood of a trajectory drawn from a conditional MP  $X$  can be written similar to Equation 2.11,

$$\begin{aligned} \Pr(X^{[0,T]} \mid \mathbf{Q}_{X \mid \mathbf{U}}) &= \left( \prod_{\mathbf{u}} \prod_i q_{i \mid \mathbf{u}}^{M[x_i \mid \mathbf{u}]} \exp(-q_{i \mid \mathbf{u}} T[x_i \mid \mathbf{u}]) \right) \left( \prod_{\mathbf{u}} \prod_i \prod_{j \neq i} \left( \frac{q_{i,j \mid \mathbf{u}}}{q_{i \mid \mathbf{u}}} \right)^{M[x_i, x_j \mid \mathbf{u}]} \right) \\ &= \prod_{\mathbf{u}} \prod_{j \neq i} \exp(-q_{i,j \mid \mathbf{u}} T[x_i \mid \mathbf{u}]) q_{i,j \mid \mathbf{u}}^{M[x_i, x_j \mid \mathbf{u}]} \end{aligned} \quad (2.14)$$

with the sufficient statistics introduced in Section 2.2.1.1 are also conditioned on parent nodes.

### 2.2.2. The CTBN Model

Evidently, a homogenous CTMP can be considered as a conditional MP whose set of parents is empty. Thus, a CTBN can be formed as a set of conditional Markov processes.

Let  $\mathbf{X}$  be a CTBN with local variables  $X_n$ ,  $n \in \{1, \dots, N\}$ , each with a state space  $\mathcal{X}_n$ . Given the dependencies of each variable as a set of its parents  $\mathbf{U}_n = \text{Par}(X_n)$ , the transition model of each local variable  $X_n$  is modelled as conditional Markov processes. [2] In the following, the set of all conditional transition intensity matrices are denoted as  $\mathbf{Q}$ .

Consider a trajectory drawn from  $\mathbf{X}$ , such that  $\mathbf{X}^{[0,T]} = \{X_1^{[0,T]}, X_2^{[0,T]}, \dots, X_N^{[0,T]}\}$ . Following Equation 2.14, the likelihood of this trajectory can be written as follows.

$$\Pr(\mathbf{X}|\mathbf{Q}) = \prod_{n=1}^N \prod_{\mathbf{u} \in \mathbf{U}_n} \prod_{x_i \in \mathcal{X}_n} \prod_{x_j \in \mathcal{X}_n \setminus x_i} \exp\left(q_{i,j|\mathbf{u}}^n T_n[x_i | \mathbf{u}]\right) (q_{i,j|\mathbf{u}}^n)^{M_n[x_i, x_j | \mathbf{u}]} \quad (2.15)$$

where  $T_n[\cdot]$  and  $M_n[\cdot]$  indicates the sufficient statistics for  $X_n$ .

## 2.3. Belief State in Partially Observable Markov Decision Processes

Partially observable Markov decision process (POMDP) framework provides a model of an agent which interacts with its environment but is unable to obtain certain information about its state. Instead, the agent gets an observation which is a function of the true state, e.g. noisy observations, translation. The main goal, similar to Markov decision processes (MDPs), is to learn a policy solving a task by optimizing a reward function. The problem of decision making under uncertainty can be decomposed into two parts for the agent. The first is to keep a belief state which summarizes past experiences, and the second is to optimize a policy which will give an action based on the belief state. [3, 4]

The belief state, if represented as a probability distribution over states, provides a sufficient statistics over the agent's past experiences.

In the problem considered in this thesis, the agent node  $X_3$  cannot observe the incoming messages directly, rather a summary of them. This presents a POMDP problem. However, since the optimal policy of the agent is assumed to be given, the theory for policy optimization is skipped.

In the following, update methods for belief state are introduced, where belief state refers to the posterior probability distribution over the environment states.

### 2.3.1. Exact/Bayes(?) Belief State Update

Consider a POMDP problem, with discrete state space  $S$ , action space  $A$ , observation space  $\Omega$ . In a scenario where a compact representation of the *transition model*,  $T(s, a, s')$ , and *observation model*,  $O(s', a, o)$ , is available, the belief state update can be obtained via Bayes' theorem [3]:

$$\begin{aligned}
b'(s') &= \Pr(s'|o, a, b) \\
&= \frac{\Pr(o|s', a, b) \Pr(s'|a, b)}{\Pr(o|a, b)} \\
&= \frac{\Pr(o|s', a) \sum_{s \in S} \Pr(s'|a, b, s) \Pr(s|a, b)}{\Pr(o|a, b)} \\
&= \frac{O(s', a, o) \sum_{s \in S} T(s, a, s') b(s)}{\Pr(o|a, b)}
\end{aligned} \tag{2.16}$$

### 2.3.2. Filtering for CTMP

Equation 2.16 is discrete-time solution of belief state. However, since in the model described in Section 2.1, the parent nodes are modelled as CTMPs, thus the environment state for the agent is the state of a CTMP, the belief state should be solved in continuous-time. This is achieved by the inference of the posterior probability of CTMP. [5]

*Filtering problem* in statistical context, as opposed to deterministic digital filtering, refers to the inference of the conditional probability of the true state of the system at some point in time, given the history of observations. [6]

Let  $X$  be a CTMP with transition intensity matrix  $\mathbf{Q}$ . Assume discrete-time observations denoted by  $y_1 = y(t_1), \dots, y_N = y(t_N)$ . The belief state can be written as:

$$b(x_i; t_N) = \Pr(X(t_N) = x_i \mid y_1, \dots, y_N) \tag{2.17}$$

From the master equation given in Equation 2.5, it follows that:

$$\frac{d}{dt} b(x_j; t) = \sum_{\forall i} q_{i,j} \cdot b(x_i; t) \tag{2.18}$$

The time-dependent belief state  $b(t)$  is a row vector with  $\{b(x_i; t)\}_{x_i \in \mathcal{X}}$ . This posterior probability can be described by a system of ODEs:

$$\frac{db(t)}{dt} = b(t)\mathbf{Q} \tag{2.19}$$

where the initial condition  $b(0)$  is row vector with  $\{b(x_i; t)\}_{x_i \in \mathcal{X}}$  [5]. The solution to this ODE is

$$b(t) = b(0) \exp(t\mathbf{Q}). \tag{2.20}$$



The belief state update at discrete times of observation  $y_t$  is derived as

$$\begin{aligned}
b(x_i; t_N) &= \Pr(X(t_N) = x_i, | y_1, \dots, y_N) \\
&= \frac{\Pr(y_1, \dots, y_N, X(t_N) = x_i)}{\Pr(y_1, \dots, y_N)} \\
&= \frac{\Pr(y_N | y_1, \dots, y_{N-1}, X(t_N) = x_i)}{\Pr(y_N | y_1, \dots, y_{N-1})} \frac{\Pr(y_1, \dots, y_{N-1}, X(t_N) = x_i)}{\Pr(y_1, \dots, y_{N-1})} \\
&= Z_N^{-1} \Pr(y_N | X(t_N) = x_i) \Pr(X(t_N) = x_i | y_1, \dots, y_{N-1}) \\
&= Z_N^{-1} \Pr(y_N | X(t_N) = x_i) b(x_i; t_N^-)
\end{aligned} \tag{2.21}$$

where  $Z_N = \sum_{x_i \in \mathcal{X}} \Pr(y_N | X(t_N) = x_i) b(x_i; t_N^-)$  is the normalization factor [5].

### 2.3.3. Belief State Update using Particle Filter

In a more realistic scenario, the exact update of belief state may not be feasible for several reasons. The computation of Bayes belief update is expensive for large state spaces. Moreover, a problem with continuous state spaces requires a belief state represented as probability distributions over infinite state space rather than a collection of probabilities as given in Sec.2.3.1. [7] Another reason could be the lack of a compact representation of transition or observation models. Under such circumstances, the belief state is obtained using sample-based approximation methods. [7]

It should be noted that since the belief state is nothing but the conditional probability of true states given the observations, the problem at hand poses a filtering problem as described in Section 2.3.2.

#### 2.3.3.1. Particle Filtering

Particle filtering is one of the most commonly used Sequential Monte Carlo (SMC) algorithms. The popularity of this method thrives from the fact that, unlike other approximation methods such as Kalman Filter, it does not assume a linear Gaussian model. This advantage offers great flexibility and finds application in a wide range of areas.[8]

The key idea in particle filtering is to approximate a target distribution  $p(x)$  by a set of samples (particles) drawn from that distribution. This is achieved by sequentially updating the particles through two steps. The first step is *importance sampling*. Since the target distribution is not available, the particles are generated from a *proposal distribution*  $q(x)$  and weighted according to the difference between target and proposal distributions. The second step is to resample the particles using these weights with replacement. [6]

**QUESTION: I wrote the formulas as well but thought they may not be very relevant. Should i include them?**

In this application, the particles to represent the belief state are drawn from marginalized

CTBN. The algorithm for belief state update through particle filtering and marginal process is given in the following chapter.

### 2.3.3.2. Marginalized Continuous-Time Markov Processes

Let  $\mathbf{X}$  be a CTBN with local variables  $X_n$ ,  $n \in \{1, \dots, N\}$ , and set of conditional intensity matrices  $\mathbf{Q}$ . In the following, it is assumed that every non-diagonal entry in  $\mathbf{Q}_n | \mathbf{u}$  is Gamma distributed with shape and rate parameters,  $\alpha_{i,j|\mathbf{u}}^n$  and  $\beta_{i,j|\mathbf{u}}^n$ .

The marginal process description of  $\mathbf{X}$  considering a single trajectory in the interval  $[0, t)$  is given as follows:

$$\begin{aligned} \Pr(X_n(t+h) = x_j | X_n(t) = x_i, \mathbf{U}_n(t) = \mathbf{u}, \mathbf{X}^{[0,t)}) \\ = \int \Pr(X_n(t+h) = x_j | X_n(t) = x_i, \mathbf{U}_n(t) = \mathbf{u}, Q_{n|\mathbf{u}}, \mathbf{X}^{[0,t)}) p(Q_{n|\mathbf{u}}) dQ_{n|\mathbf{u}} \\ = \delta_{i,j} + \mathbb{E}[q_{i,j|\mathbf{u}}^n | \mathbf{X}^{[0,t]} = \mathbf{x}^{[0,t]}] h + o(h), \end{aligned} \quad (2.22)$$

By integrating out the intensity matrix  $Q_{n|\mathbf{u}}$ , the parameter is replaced by its expected value given the history of the process. It should be noted that by doing so, the process becomes parameter-free, and thus self-exciting.

The derivation of the conditional expectation for marginal CTBN follows from the Bayes' rule:

$$p(\mathbf{Q} | \mathbf{X}_{[0,t]}) = \frac{p(\mathbf{X}_{[0,t]} | \mathbf{Q}) p(\mathbf{Q})}{p(\mathbf{X}_{[0,t]})} \quad (2.23)$$

Equation 2.23, written for single trajectory  $\mathbf{X}_{[0,t]}$ , can be extended for multiple trajectories. Consider  $K$  trajectories drawn from CTBN  $\mathbf{X}$ , denoted by  $\xi_t = \{\mathbf{X}^{[0,t],1}, \mathbf{X}^{[0,t],2}, \dots, \mathbf{X}^{[0,t],K}\}$ . Since the trajectories are conditionally independent, given  $\mathbf{Q}$ , using Equation 2.15 the likelihood of set  $\xi_t$  is written as,

$$\Pr(\xi_t | \mathbf{Q}) = \prod_{n=1}^N \prod_{\mathbf{u} \in \mathbf{U}_n} \prod_{x_i \in \mathcal{X}_n} \prod_{x_j \in \mathcal{X}_n \setminus x_i} \exp(q_{i,j|\mathbf{u}}^n T_n[x_i | \mathbf{u}]) (q_{i,j|\mathbf{u}}^n)^{M_n[x_i, x_j | \mathbf{u}]} \quad (2.24)$$

where the joint sufficient statistics of  $X_n$  over all  $K$  trajectories are denoted by  $T_n[x_i | \mathbf{u}] = \sum_{k=1}^K T_n^k[x_i | \mathbf{u}]$  and  $M_n[x_i, x_j | \mathbf{u}] = \sum_{k=1}^K M_n^k[x_i, x_j | \mathbf{u}]$ .

Given independent Gamma-priors on transition intensities, the expectation in Equation 2.22 can be evaluated as follows:

$$\mathbb{E}[q_{i,j|\mathbf{u}}^n | \xi_t] = \frac{\alpha_{i,j|\mathbf{u}}^n + M_n[x_i, x_j | \mathbf{u}]}{\beta_{i,j|\mathbf{u}}^n + T_n[x_i | \mathbf{u}]} \quad (2.25)$$

**TODO:** when is this parameter updated? explain. write about the dependency problem, how is this okay?

## 2.4. Sampling Algorithms

### 2.4.1. Gillespie Algorithm for Generative CTBN

Gillespie algorithm is a computer-oriented Monte Carlo simulation procedure that is originally proposed to simulate the reactions of molecules in any spatially homogeneous chemical system. Such systems are regarded as Markov processes and represented via their master equations, which cannot be directly used to obtain realizations of the process. Gillespie algorithm is an efficient tool to overcome this problem. [9]

This algorithm can also be applied to sample *events* from a CTBN given the transition intensity matrices, where an event refers to a transition occurring at a specific point in time. This procedure is introduced as *Generative CTBN* in [2].

---

**Algorithm 1:** Generative CTBN

---

**Input** : Structure of the network with N local variables  $X_1, X_2, \dots, X_n$  with state-space  $\chi_n = \{x_1, \dots, x_m\}$   
Transition intensity matrices  $\mathbf{Q}_n$  with entries  $q_{i,j}^n$   
 $T_{max}$  to terminate simulation

**Output** : Sample trajectory of the network

**Initialize:** Initialize node values  $X_n(0) = x_i \in \chi_n$

- 1: **while**  $t < T_{max}$  **do**
- 2:    $\tau \sim \exp(\sum_{\forall n} \sum_{\forall i \neq j} q_{i,j}^n)$
- 3:   transitioning node is randomly drawn with probability  $P(X_n) = \frac{q_i^n}{\sum_{\forall n} q_i^n}$
- 4:   next state is randomly drawn with probability  $P(x_j) = \frac{q_{i,j}^n}{q_i^n}$
- 5:    $t \leftarrow t + \tau$
- 6: **end while**

---

### 2.4.2. Thinning Algorithm

Thinning algorithm is a method introduced to simulate nonhomogenous Poisson processes. [10] Later, it is adapted to sample from Hawkes processes, a self-exciting process with time-dependent intensity function. [11, 12] This algorithm is used here to simulate inhomogeneous

Markov process.

---

**Algorithm 2:** Thinning Algorithm

---

**Input** :  $\lambda(t)$  the intensity function of the inhomogenous process

$N$  number of events to terminate simulation

**Output** : Sample trajectory of the process

**Initialize:** Time  $t = 0$

1: **while**  $i < N$  **do**

2:   the upper bound for intensity,  $\lambda^*$

3:   transition time  $\tau$  drawn by  $u \sim U(0, 1)$  and  $\tau = \frac{-\ln(u)}{\lambda^*}$

4:    $t \leftarrow t + \tau$

5:   draw  $s \sim U(0, 1)$

6:   **if**  $s \leq \frac{\lambda(t)}{\lambda^*}$  **then**

7:     sample accepted and  $t_i = t, i = i + 1$

8:   **end if**

9: **end while**

---

## 3. Experimental Setup

This chapter presents the methodology used in this thesis. First, it is explained how different frameworks introduced in Chapter 2 are put into use. Then, the algorithms used in data generation and inference are given in detail. The results from these experiments are presented in the succeeding chapter.

### 3.1. The Model

A detailed graphical model explored in this thesis is given in the Section 3.1. This model presents an intersection of continuous-time Bayesian network and partially observable Markov decision process frameworks.

- The transition models of the nodes  $X_1, X_2$  and  $X_3$ , and the dependencies between them are modelled as CTBN.
- The interaction of agent node  $X_3$  and its environment is modelled as POMDP.

#### 3.1.1. CTBN Model

The transition models of the nodes and the dependencies between them are modelled as continuous-time Bayesian network (CTBN), denoted by  $\mathbf{X}$ . The network  $\mathbf{X}$  represents a stochastic process over a structured multivariate state space  $\mathcal{X} = [\mathcal{X}_1, \mathcal{X}_2, \mathcal{X}_3]$ .

The parent nodes  $X_1$  and  $X_2$  emit their states as messages. The dynamics of these nodes are modelled as independent homogeneous continuous-time Markov processes  $X_i(t)$ , with binary-valued states  $\mathcal{X}_i = \{0, 1\}$  for  $i \in \{1, 2\}$ . These processes are defined by transition intensity matrices  $\mathbf{Q}_i$ , which are assumed to be Gamma distributed with shape and rate parameters  $\boldsymbol{\alpha} = [\alpha_0, \alpha_1]$  and  $\boldsymbol{\beta} = [\beta_0, \beta_1]$ , respectively, and are in the following

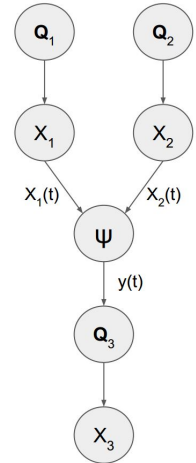


Figure 3.1.: Hierarchical model.

forms.

$$\mathbf{Q}_i = \begin{bmatrix} -q_0^i & q_0^i \\ q_1^i & -q_1^i \end{bmatrix} \quad (3.1)$$

$$\mathbf{Q}_i \sim \text{Gam}(\boldsymbol{\alpha}^i, \boldsymbol{\beta}^i) \text{ for } i \in \{1, 2\} \quad (3.2)$$

It should be noted that in Equation 3.1, the suffixes are simplified using the fact that  $q_i = \sum_{i \neq j} q_{i,j}$ .

The agent  $X_3$  is modelled as inhomogenous continuous-time Markov process with binary states  $X_3 = \{0, 1\}$  and set of actions  $a \in \{a_0, a_1\}$ , and set of transition intensity matrices which contains one matrix corresponding to each action,  $\mathbf{Q}_{3|a} = \{\mathbf{Q}_{3|a_0}, \mathbf{Q}_{3|a_1}\}$ .

The dependencies are represented by set of parents for each node  $\mathbf{U}_n = \text{Par}(X_n)$  and for the model shown in Section 3.1 can be written as follows:

$$\begin{aligned} \mathbf{U}_1, \mathbf{U}_2 &= \emptyset \\ \mathbf{U}_3 &= \{X_1, X_2\} \end{aligned}$$

### 3.1.2. POMDP Model

In a conventional POMDP scenario, there are two problems to be addressed, one is belief state update and the other is policy optimization. As mentioned in Section 2.3, in the problem at hand, the policy of agent  $X_3$  is assumed to be optimal and given. Thus, the POMDP model of the agent only consists of belief state update. A detailed view of the agents interaction with its environment from POMDP framework perspective is given in the Figure 3.2.

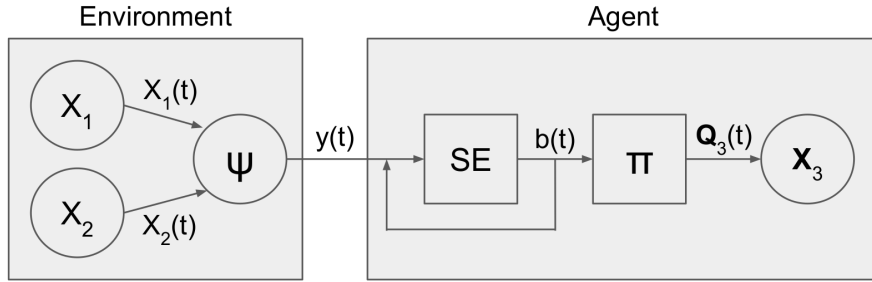


Figure 3.2.: Closer look to agent-environment interaction from the perspective of POMDP framework.

It should be noted that, the interaction in Figure 3.2 is only one-sided, the state or action of the agent does not affect the environment.

### 3.1.2.1. Observation Model

The messages sent by the parent nodes are translated by the observation model. The agent node  $X_3$  does not have a direct access to the messages, but observes a translation of them. The observation model gives a probability distribution over the observation for each combination of parent messages.

$$\psi(x_1, x_2) = \Pr(y(t) \mid X_1(t), X_2(t)) \quad (3.3)$$

### 3.1.2.2. Belief State

The *state estimator* (labelled as SE in Figure 3.2) forms a belief over the parent states, denoted by  $b(x_1, x_2; t)$ .

$$b(x_1, x_2; t) = \Pr(X_1(t) = x_1, X_2(t) = x_2 \mid y_1, \dots, y_t) \quad (3.4)$$

**Exact Belief State Update** Given the transition intensity matrices of parent nodes,  $\mathbf{Q}_1$  and  $\mathbf{Q}_2$ , the continuous-time belief state update poses a filtering problem for CTMPs (Section 2.3.2).

Consider a subsystem of CTBN model, consisting of only the parent nodes,  $X_1$  and  $X_2$ . These two processes can be represented as one single *joint* process,  $X_P$ , with multivariate state space  $\chi_P = \{(x_1, x_2)\}_{x_1 \in \chi_1, x_2 \in \chi_2} = \{(0, 0), (0, 1), (1, 0), (1, 1)\}$ . The transition intensity matrix of the new joint system,  $\mathbf{Q}_P$  is obtained by amalgamation operation between  $\mathbf{Q}_1$  and  $\mathbf{Q}_2$  (see Appendix A) [2].

$$\mathbf{Q}_P = \mathbf{Q}_1 * \mathbf{Q}_2 \quad (3.5)$$

Then, the filtering problem can be formulated according to this joint process of parents.

$$b(x_p; t) = \Pr(X_P(t) = x_p \mid y_1, \dots, y_t) \quad (3.6)$$

Consider discrete-time observations from this process, denoted by  $y_1 = y(t_1), \dots, y_N = y(t_N)$  and time-dependent belief state  $b(t)$  as a row vector with  $\{b(x_p; t)\}_{x_p \in \chi_P}$ . Following Equation 2.20 and Equation 2.21, the belief state update is evaluated as

$$b(t) = b(0) \exp(t\mathbf{Q}_P) \quad (3.7)$$

with the initial condition  $b(0)$ . The update at discrete times of observation  $y_t$  is

$$b(x_p; t_N) = Z_N^{-1} \Pr(y_N \mid X_P(t_N) = x_p) b(x_p; t_N^-) \quad (3.8)$$

$$= Z_N^{-1} \psi(x_p) b(x_p; t_N^-) \quad (3.9)$$

where  $Z_N = \sum_{x_p \in \chi_P} \psi(x_p) b(x_p; t_N^-)$  is the normalization factor.

**Belief State Update Using Particle Filter** The assumption that full information of parent dynamics being available is unrealistic. In an environment as described above, the agent most probably doesn't have access to the parameters  $\mathbf{Q}_1$  and  $\mathbf{Q}_2$ , rather may have some prior beliefs over them. Thus, in order to simulate a more realistic model and be able to marginalize out these parameters from inference problem, the joint process (introduced in previous section) is replaced with its marginalized counterpart. Using the Gamma-priors over  $\mathbf{Q}_1$  and  $\mathbf{Q}_2$  (Equation 3.2) and sufficient statistics over the particle history, the particles are drawn from this marginalized process. With every new observation, the particles are propagated through the process, while the sufficient statistics are updated and the parameters are re-estimated after each particle using the Equation 2.25. The belief state then obtained as the distribution of states over the particles,

$$b(x_p; t) = \frac{1}{N} \sum_{i=1}^N \delta_{p_i(t), x_p} \quad (3.10)$$

where  $N$  is the number of particles,  $\mathbf{p}$  is the set of particles, and  $\delta$  is the Kronecker delta.

---

**Algorithm 3:** Marginal particle filter for belief state update

---

**Input** : Observation  $y_k$  at time  $t_k$ , set of particles  $\mathbf{p}^{k-1}$ , estimated  $\hat{Q}$   
**Output:** New set of particles  $\mathbf{p}^k$ ,  $\mathbf{b}^{[t_{k-1}, t_k]}$

- 1: **for**  $p_m \in \mathbf{p}^{k-1}$  **do**
- 2:    $p_m = \{x_m, \hat{Q}\} \leftarrow \text{Propagate particle through marginal process from } t_{k-1} \text{ to } t_k$
- 3:    $\hat{Q} \leftarrow \text{sufficient statistics added from } p_m[t_{k-1}, t_k]$   
       // observation likelihood assigned as particle weight
- 4:    $w_m \leftarrow p(y_k \mid X_P(t_k) = x_m)$
- 5: **end for**  
       // belief state from  $t_{k-1}$  to  $t_k$
- 6:  $\mathbf{b}^{[t_{k-1}, t_k]} \leftarrow \left\{ \frac{1}{N} \sum_{i=1}^N \delta_{p_i^{[t_{k-1}, t_k]}, x_p} \right\}_{x_p \in \mathcal{X}_P}$   
       // normalize weights
- 7:  $w_m \leftarrow \frac{w_m}{\sum_m w_m}$   
       // resample particles
- 8: **for**  $p_m \in \mathbf{p}_k$  **do**
- 9:    $p_m \leftarrow \text{Sample from } p_k \text{ with probabilities } w_m \text{ with replacement}$
- 10: **end for**

---

### 3.1.2.3. Optimal Policy

The optimal policy is defined using a polynomial function of belief state.

$$\pi(b) = \begin{cases} a_0 & \text{if } \mathbf{w}b^\top > 0.5 \\ a_1 & \text{otherwise} \end{cases} \quad (3.11)$$

where  $\mathbf{w}$  is a row vector of weights.



Given the optimal policy,  $\pi(b)$ , the agent takes an action based on the belief state. In the setting described above, taking an action means to change its internal dynamics to the transition intensity matrix corresponding to that action.

$$a(t) = \pi(b(t)) \quad (3.12)$$

$$\mathbf{Q}_3(t) = \begin{cases} \mathbf{Q}_{3|a_0} & \text{if } a(t) = a_0 \\ \mathbf{Q}_{3|a_1} & \text{otherwise} \end{cases} \quad (3.13)$$

## 3.2. Inference of Observation Model

Inference problem is considered for deterministic observation models. Considering the number of states of parents and the observations, there are a number of possible observation models. Given the parent trajectories, the belief state and the resulting  $\mathbf{Q}_3$  trajectory is computed for each observation model. Then the likelihood of  $X_3$  trajectory given these  $\mathbf{Q}_3$  trajectories are compared for maximum likelihood estimation.

$$\hat{\psi} = \operatorname{argmax} \Pr(X_3^{[0,T]} | Q_3^{[0,T]}) \quad (3.14)$$

**QUESTION:** Is this okay to put results for 3 observation models or should i run the results for all 81 observation models we have? The problem is there are some observation models that we wouldn't be able to distinguish

### 3.2.1. Likelihood Model

Consider a trajectory in the dataset, denoted by  $\mathbf{X}^{[0,T]} = \{X_1^{[0,T]}, X_2^{[0,T]}, X_3^{[0,T]}\}$ . The set of parameters to the system, as introduced before, is written as  $\theta = \{\mathbf{Q}_1, \mathbf{Q}_2, \pi, \psi\}$ . Then likelihood of the sample trajectory  $\mathbf{X}^{[0,T]}$  can be written as:

$$\begin{aligned} P(\mathbf{X}^{[0,T]} | \theta) &= P(X_1^{[0,T]}, X_2^{[0,T]}, X_3^{[0,T]} | \mathbf{Q}_1, \mathbf{Q}_2, \pi, \psi) \\ &= P(X_3^{[0,T]} | X_1^{[0,T]}, X_2^{[0,T]}, \mathbf{Q}_1, \mathbf{Q}_2, \pi, \psi) P(X_1^{[0,T]} | \mathbf{Q}_1) P(X_2^{[0,T]} | \mathbf{Q}_2) \\ &= P(X_3^{[0,T]} | X_1^{[0,T]}, X_2^{[0,T]}, \pi, \psi) P(X_1^{[0,T]} | \mathbf{Q}_1) P(X_2^{[0,T]} | \mathbf{Q}_2) \\ &= P(X_3^{[0,T]} | \mathbf{Q}_3^{[0,T]}) P(X_1^{[0,T]} | \mathbf{Q}_1) P(X_2^{[0,T]} | \mathbf{Q}_2) \end{aligned} \quad (3.15)$$

As mentioned before, it is plausible to marginalize out the parameters  $\mathbf{Q}_1$  and  $\mathbf{Q}_2$ , for a more realistic model and inference. Noting that in case the belief state is updated using filtering of CTMPs (See Section 3.1.2.2),  $\mathbf{Q}_3^{[0,T]}$  becomes a deterministic function of all the parameters including  $\mathbf{Q}_1$  and  $\mathbf{Q}_2$ , the marginalization cannot be carried out analytically on Equation 3.15. On the other hand, particle filtering removes this dependency on  $\mathbf{Q}_1$  and  $\mathbf{Q}_2$  by using marginalized counterpart of CTMPs (See Section 3.1.2.2), leaving it straightforward to marginalize out the parameters on Equation 3.15.

Marginalizing the likelihood over  $Q_1$  and  $Q_2$ :

$$\begin{aligned}
P(\mathbf{X}^{[0,T]} \mid \pi, \psi) &= \int \int P(\mathbf{X}^{[0,T]} \mid \theta) P(\mathbf{Q}_1) P(\mathbf{Q}_2) d\mathbf{Q}_1 d\mathbf{Q}_2 \\
&= \int \int P(X_3^{[0,T]} \mid \mathbf{Q}_3^{[0,T]}) P(X_1^{[0,T]} \mid \mathbf{Q}_1) P(X_2^{[0,T]} \mid \mathbf{Q}_2) P(\mathbf{Q}_1) P(\mathbf{Q}_2) d\mathbf{Q}_1 d\mathbf{Q}_2 \\
&= P(X_3^{[0,T]} \mid \mathbf{Q}_3^{[0,T]}) \int P(X_1^{[0,T]} \mid \mathbf{Q}_1) P(\mathbf{Q}_1) d\mathbf{Q}_1 \int P(X_2^{[0,T]} \mid \mathbf{Q}_2) P(\mathbf{Q}_2) d\mathbf{Q}_2
\end{aligned} \tag{3.16}$$

Marginalized likelihood function for binary-valued homogenous CTMP is derived in Appendix B.

Plugging Equation B.2 in Equation 3.16 for both  $X_1$  and  $X_2$ :

$$\begin{aligned}
P(\mathbf{X}^{[0,T]} \mid \pi, \Phi) &= P(X_3^{[0,T]} \mid \mathbf{Q}_3^{[0,T]}) \prod_{x_1 \in \{0,1\}} \frac{\beta_{x_1}^{\alpha_{x_1}}}{\Gamma(\alpha_{x_1})} (T_{x_1} + \beta_{x_1})^{M_{x_1} + \alpha_{x_1}} \Gamma(M_{x_1} + \alpha_{x_1}) \\
&\quad \prod_{x_2 \in \{0,1\}} \frac{\beta_{x_2}^{\alpha_{x_2}}}{\Gamma(\alpha_{x_2})} (T_{x_2} + \beta_{x_2})^{M_{x_2} + \alpha_{x_2}} \Gamma(M_{x_2} + \alpha_{x_2})
\end{aligned} \tag{3.17}$$

### 3.3. Data Generation

The dataset contains a number of trajectories drawn from CTBN  $\mathbf{X}$ . Following the notation in Chapter 2,  $K$  trajectories in time interval  $[0, T]$  are denoted by  $\xi_T = \{\mathbf{X}^{[0,T],1}, \mathbf{X}^{[0,T],2}, \dots, \mathbf{X}^{[0,T],K}\}$ , where  $\mathbf{X}^{[0,T],k} = \{X_1^{[0,T],k}, X_2^{[0,T],k}, X_3^{[0,T],k}\}$  denotes a single trajectory for all nodes. Every trajectory comprises of state transitions in the interval, and the times of these transitions.

#### 3.3.1. Sampling Algorithm

In order to sample trajectories from CTBN, two sampling algorithms introduced in Section 2.4 are combined. Gillespie algorithm is used to sample from the parent nodes,  $X_1$  and  $X_2$ , while thinning algorithm is applied to overcome the challenges that come with conditional intensity matrix of the agent,  $X_3$ . It should be noted that, Algorithm 1 is applicable to any nodes in a CTBN, both homogenous and conditional MPs. However, since in this setting, the intensity matrix is conditioned on the belief state and the policy, instead of directly on the parent states, a more general algorithm suitable for inhomogenous MPs, thinning algorithm,

is preferred. Algorithm 4 describes the procedure to draw samples using particle filtering.

---

**Algorithm 4:** Sampling trajectories with particle filtering

---

**Input** : Gamma-prior parameters on parents' transition intensity matrices  $\alpha^1, \beta^1, \alpha^2, \beta^2$   
Set of agent's transition intensity matrices  $\mathbf{Q}_3$   
 $T_{max}$  to terminate simulation

**Output** : Sample trajectory of the network

**Initialize:** Sample  $\mathbf{Q}_1$  and  $\mathbf{Q}_2$  from their priors  
Initialize nodes uniformly  $X_n(0) = x_i \in \mathcal{X}_n$   
Initialize particles uniformly  $p^i(0) = x_p \in \mathcal{X}_P$   
 $t = 0$

- 1: **while**  $t < T_{max}$  **do**
- 2:   Draw next transition for  $X_1$  and  $X_2$  ( $\tau_{parent}$ ,  $x_1$  and  $x_2$  using Algorithm 1)
- 3:    $t_{parent} \leftarrow t + \tau_{parent}$  // **transition time for parents**
- 4:    $y_{t_{parent}} \sim \psi(x_1, x_2)$  // **new observation at  $t_{parent}$**
- 5:   Update particle filter and obtain  $\mathbf{b}^{[t, t_{parent}]}$
- 6:    $a^{[t, t_{parent}]} \leftarrow \pi(\mathbf{b}^{[t, t_{parent}]})$
- 7:    $\mathbf{Q}_3^{[t, t_{parent}]} \leftarrow \mathbf{Q}_3|_{a^{[t, t_{parent}]}}$
- 8:    $t_{agent} \leftarrow t$
- 9:   **while**  $t_{agent} < t_{parent}$  **do**
- 10:     the upper bound for intensity,  $q_3^*$ <sup>1</sup>
- 11:     transition time  $\tau_{agent}$  drawn by  $u \sim U(0, 1)$  and  $\tau_{agent} = \frac{-\ln(u)}{q_3^*}$
- 12:      $t_{agent} \leftarrow t_{agent} + \tau_{agent}$
- 13:     draw  $s \sim U(0, 1)$ , accept transition if  $s \leq \frac{q_3(t_{agent})}{q_3^*}$
- 14:   **end while**
- 15:    $t \leftarrow t_{parent}$
- 16: **end while**

---

### 3.4. Configurations

The configurations given below are used for the results presented in the following chapter, if not specified otherwise.

- Gamma priors for parent dynamics such that  $\mathbf{Q}_i \sim \text{Gam}(\boldsymbol{\alpha}^i, \boldsymbol{\beta}^i)$  for  $i \in \{1, 2\}$ , and  $\boldsymbol{\alpha} = [\alpha_0, \alpha_1]$  and  $\boldsymbol{\beta} = [\beta_0, \beta_1]$

$$\boldsymbol{\alpha}^1 = [5, 10], \boldsymbol{\beta}^1 = [5, 20] \tag{3.18}$$

$$\boldsymbol{\alpha}^2 = [20, 20], \boldsymbol{\beta}^2 = [20, 10] \tag{3.19}$$

---

<sup>1</sup> $q$  is the transition intensity associated with the current state of the agent.

- Transition intensity matrices of  $X_1$  and  $X_2$  sampled from priors given above

$$\mathbf{Q}_1 = \begin{bmatrix} -1.117 & 1.117 \\ 0.836 & -0.836 \end{bmatrix} \quad (3.20)$$

$$\mathbf{Q}_2 = \begin{bmatrix} -1.077 & 1.077 \\ 2.324 & -2.324 \end{bmatrix} \quad (3.21)$$

- State space,  $S = \mathcal{X}_P = \{(0, 0), (0, 1), (1, 0), (1, 1)\}$
- Observation space,  $O = \{0, 1, 2\}$
- Action space,  $A = \{a_0, a_1\} = \{0, 1\}$
- The set of transition intensity matrices of  $X_3$

$$\mathbf{Q}_3 = \{\mathbf{Q}_{3|a_0}, \mathbf{Q}_{3|a_1}\} = \left\{ \begin{bmatrix} -0.5 & 0.5 \\ 2 & -2 \end{bmatrix}, \begin{bmatrix} -3 & 3 \\ 0.02 & -0.02 \end{bmatrix} \right\} \quad (3.22)$$

- Number of particles,  $N = 200$

## 4. Results

### 4.1. Simulation

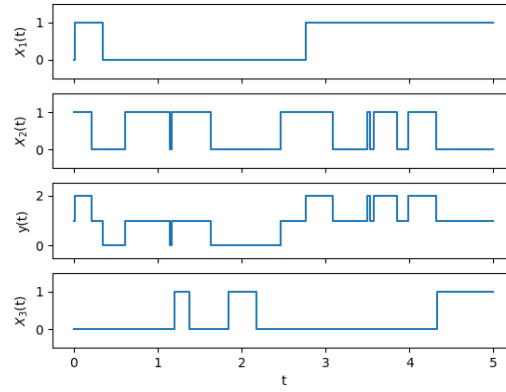


Figure 4.1.: Sampled trajectories

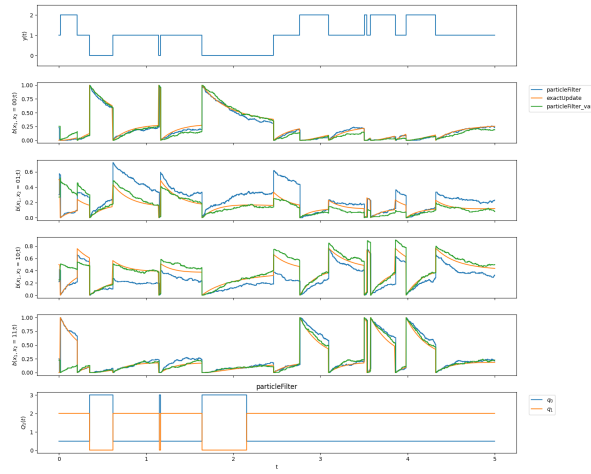


Figure 4.2.: Belief state and  $Q_3$  trajectories

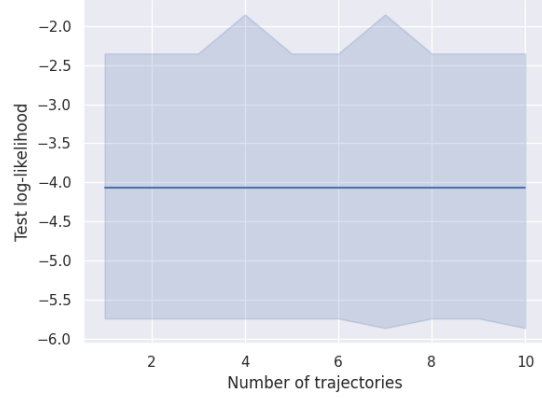
## 4.2. Results

### 4.2.1. Maximum Likelihood Estimation

$$\psi_{true} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \psi_0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \psi_1 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \psi_2 = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$



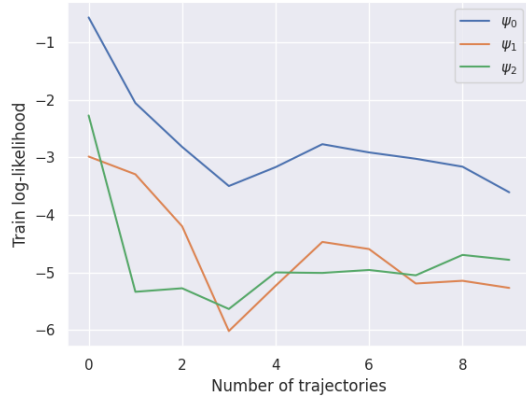
(a)



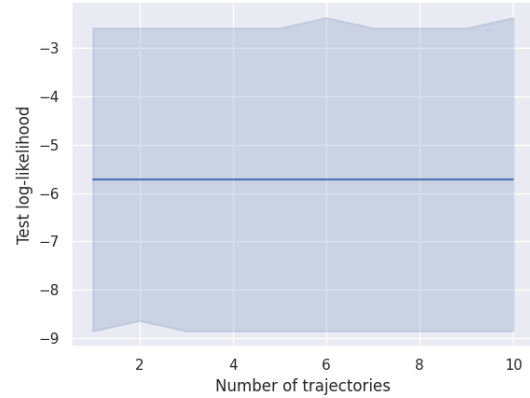
(b)

Figure 4.3.: plots of...

$$\psi_{true} = \begin{bmatrix} 0.98 & 0.01 & 0.01 \\ 0.01 & 0.98 & 0.01 \\ 0.01 & 0.98 & 0.01 \\ 0.01 & 0.01 & 0.98 \end{bmatrix}, \psi_0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \psi_1 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \psi_2 = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$



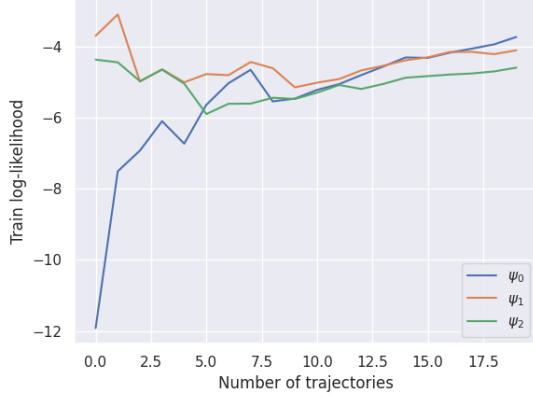
(a)



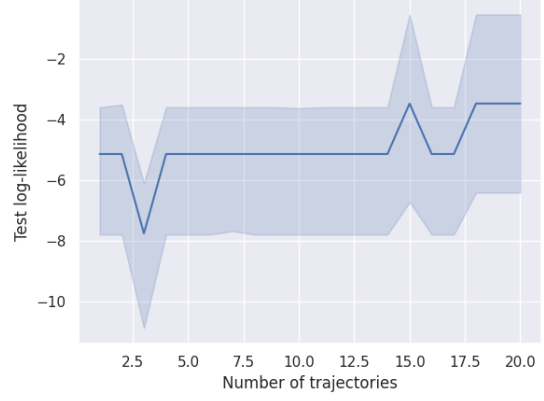
(b)

Figure 4.4.: plots of...

$$\psi_{true} = \begin{bmatrix} 0.95 & 0.025 & 0.025 \\ 0.025 & 0.95 & 0.025 \\ 0.025 & 0.95 & 0.025 \\ 0.025 & 0.025 & 0.95 \end{bmatrix}, \psi_0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \psi_1 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \psi_2 = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$



(a)



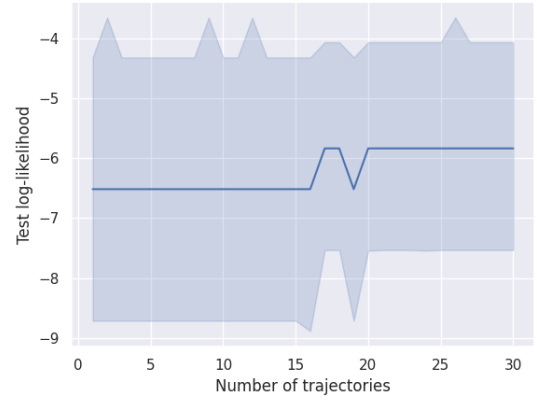
(b)

Figure 4.5.: plots of....

$$\psi_{true} = \begin{bmatrix} 0.9 & 0.05 & 0.05 \\ 0.05 & 0.9 & 0.05 \\ 0.05 & 0.9 & 0.05 \\ 0.05 & 0.05 & 0.9 \end{bmatrix}, \psi_0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \psi_1 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \psi_2 = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$



(a)



(b)

Figure 4.6.: plots of....

#### 4.2.2. Maximum Likelihood Classification

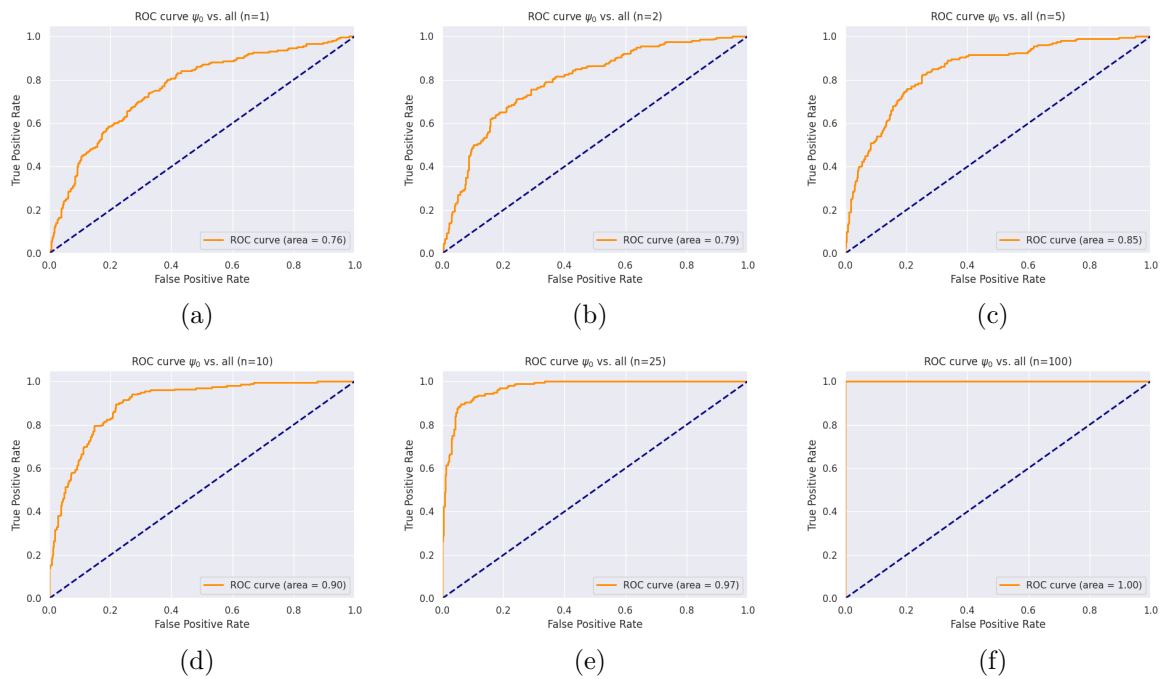


Figure 4.7.: plots of....

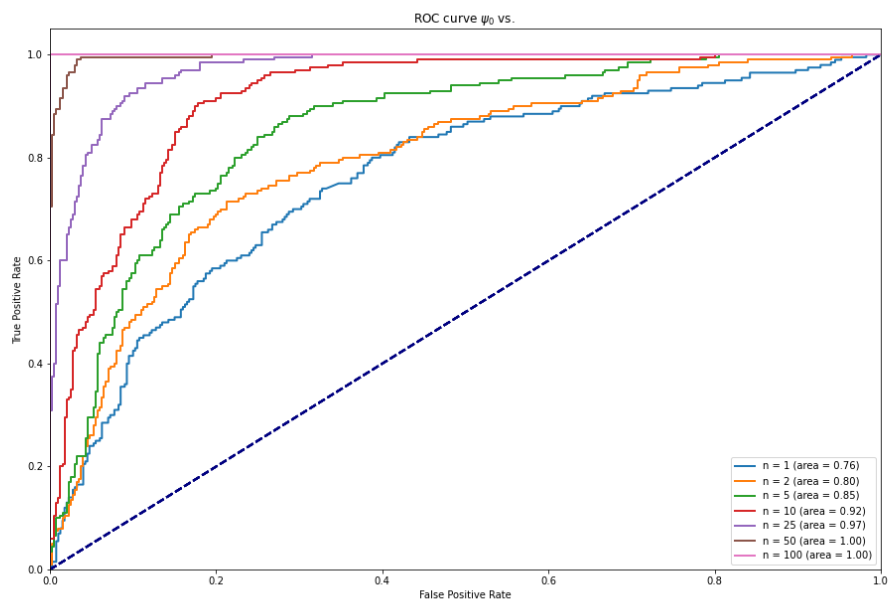


Figure 4.8.: plot of...



## **5. Conclusion**

### **5.1. Discussion**

### **5.2. Future Work**

# Bibliography

- [1] I. Cohn, T. El-Hay, N. Friedman, and R. Kupferman, “Mean Field Variational Approximation for Continuous-Time Bayesian Networks,” *Journal of Machine Learning Research*, vol. 11, pp. 2745–2783, 2010.
- [2] U. Nodelman, C. R. Shelton, and D. Koller, “Continuous Time Bayesian Networks,” 1995.
- [3] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, “Planning and acting in partially observable stochastic domains,” *Artificial Intelligence*, vol. 101, no. 1, pp. 99–134, 1998.
- [4] K. P. Murphy, “A survey of POMDP solution techniques,” *Environment*, vol. 2, no. September, p. X3, 2000.
- [5] L. Huang, L. Paulevé, C. Zechner, M. Unger, A. Hansen, and H. Koeppl, “Supporting Information for Reconstructing dynamic molecular states from single-cell time series,” 2016.
- [6] S. Godsill, “PARTICLE FILTERING: THE FIRST 25 YEARS AND BEYOND,” pp. 7760–7764, 2019.
- [7] S. Thrun, “Monte Carlo POMDPs,” vol. 40, no. 10, pp. 117–151, 1904.
- [8] A. Doucet and A. M. Johansen, “A tutorial on particle filtering and smoothing: Fifteen years later,” *Handbook of Nonlinear Filtering*, no. December, pp. 4–6, 2009.
- [9] D. T. Gillespie, “A general method for numerically simulating the stochastic time evolution of coupled chemical reactions,” *Journal of Computational Physics*, vol. 22, no. 4, pp. 403–434, 1976.
- [10] P. A. Lewis and G. S. Shedler, “Simulation of Nonhomogeneous Poisson Processes By Thinning,” *Naval research logistics quarterly*, vol. 26, no. 3, pp. 403–413, 1979.
- [11] Y. Ogaata, “On Lewis’ Simulation Method for Point Processes,” *IEEE Transactions on Information Theory*, vol. 27, no. 1, pp. 23–31, 1981.
- [12] M.-A. Rizoïu, Y. Lee, and S. Mishra, “Hawkes processes for events in social media,” *Frontiers of Multimedia Research*, pp. 191–218, 2017.

## A. Amalgamation

The amalgamation operation for two independent processes are derived as follows:

$$\begin{aligned} & \Pr(X_1(t+h) = x'_1, X_2(t+h) = x_2 \mid X_1(t) = x_1, X_2(t) = x_2) \\ &= \Pr(X_1(t+h) = x'_1 \mid X_1(t) = x_1, X_2(t) = x_2) \Pr(X_2(t+h) = x_2 \mid X_1(t) = x_1, X_2(t) = x_2) \\ &= (\delta_{x'_1, x_1} + h q_{x_1, x'_1}^1 + o(h))(1 + h q_{x_2, x_2}^2 + o(h)) \\ &= \delta_{x'_1, x_1} + h q_{x_1, x'_1}^1 + h \delta_{x'_1, x_1} q_{x_2, x_2}^2 + o(h) \end{aligned} \tag{A.1}$$

## B. Marginalized Likelihood Function for Homogenous Continuous Time Markov Processes

Let  $X$  be a homogenous CTMP. For convenience, it is assumed to be binary-valued,  $\mathcal{X} = \{x_0, x_1\}$ . The transition intensity matrix can be written in the following form:

$$\mathbf{Q} = \begin{bmatrix} -q_0 & q_0 \\ q_1 & -q_1 \end{bmatrix} \quad (\text{B.1})$$

where the transition intensities  $q_0$  and  $q_1$  are gamma-distributed with parameters  $\alpha_0, \beta_0$  and  $\alpha_1, \beta_1$ , respectively. The marginal likelihood of a sample trajectory  $X^{[0,T]}$  can be written as follows:

$$\begin{aligned} P(X^{[0,T]}) &= \int P(X^{[0,T]} \mid Q) P(Q) dQ \\ &= \int_0^\infty \left( \prod_x \exp(-q_x T_x) \prod_{x'} q_{xx'}^{M[x,x']} \right) \frac{\beta_{xx'}^{\alpha_{xx'}} q_{xx'}^{\alpha_{xx'}-1} \exp(-\beta_{xx'} q_{xx'})}{\Gamma(\alpha_{xx'})} dq_{xx'} \\ &= \prod_{i \in \{0,1\}} \int_0^\infty q_i^{M[x_i]} \exp(-q_i T[x_i]) \frac{\beta_i^{\alpha_i} q_i^{\alpha_i-1} \exp(-\beta_i q_i)}{\Gamma(\alpha_i)} dq_i \\ &= \prod_{i \in \{0,1\}} \frac{\beta_i^{\alpha_i}}{\Gamma(\alpha_i)} \int_0^\infty q_i^{M[x_i]+\alpha_i-1} \exp(-q_i(T[x_i] + \beta_i)) dq_i \\ &= \prod_{i \in \{0,1\}} \frac{\beta_i^{\alpha_i}}{\Gamma(\alpha_i)} \left( -(T_i + \beta_i)^{M[x_i]+\alpha_i} \Gamma(M[x_i] + \alpha_i, q_i(T[x_i] + \beta_i)) \right) \Big|_0^\infty \\ &= \prod_{i \in \{0,1\}} \frac{\beta_i^{\alpha_i}}{\Gamma(\alpha_i)} \left( (T[x_i] + \beta_i)^{M[x_i]+\alpha_i} \Gamma(M[x_i] + \alpha_i) \right) \end{aligned} \quad (\text{B.2})$$

where  $T_x$ , the amount of time spent in state  $x$ ,  $M[x, x']$  the number of transitions from state  $x$  to  $x'$  and  $M[x] = \sum_{x \neq x'} M[x, x']$ .