Mean Field Variational Approximations in Continuous-Time Markov Processes

A thesis submitted in partial fulfillment of the

requirements for the degree of Master of Science

by

Ido Cohn

Supervised by Prof. Nir Friedman

July 2009

The School of Computer Science and Engineering
The Hebrew University of Jerusalem, Israel

Contents

Abstract

Continuous-time Bayesian networks is a natural structured representation language for multi-component stochastic processes that evolve continuously over time. Despite the compact representation, inference in such models is intractable even in relatively simple structured networks. Here we introduce a mean field variational approximation in which we use a product of *inhomogeneous* Markov processes to approximate a joint distribution over trajectories. This variational approach leads to a globally consistent distribution, which can be efficiently queried. Additionally, it provides a lower bound on the probability of observations, thus making it attractive for learning tasks. Here I describe the theoretical foundations for the approximation, an efficient implementation that exploits the wide range of highly optimized ordinary differential equations (ODE) solvers, experimentally explore characterizations of processes for which this approximation is suitable, and show applications to a large-scale real-world inference problem.

Acknowledgements

I am deeply grateful to Nir Friedman for his supervision and guidance in this long journey that has been my Masters degree. I would also like to thank Raz Kupferman for his great help in the publication of this work. For the members of the lab I am indebted for making my Masters degree as enjoyable as humanly possible, each in their own way. Naomi for her coconut-milk dishes, Nir for his hand-made ravioli, Noa for the stuffed peppers, Ruty for her chilly, Ayelet for the couscous, Tal for his gazpacho, Assaf for his cauliflower-chicken balls, Inbar for her meatballs, Ariel for the fishballs, Ofer for the peanutbutter chicken, Aharon and Adam for their salad dicing abilities and Moran for her amazing desserts. Additionally, I wish to show my profound gratitude to Tal El-Hay, who was practically my second advisor. I am grateful for his guidance, persistance and infinite patience that enabled us to publish this work. And finally, thanks ever so much to my dear family and friends for their wholehearted support.

Chapter 1

Introduction

1.1 Motivation

Many real-life processes can be naturally thought of as evolving continuously in time. Examples cover a diverse range, including server availability, modeling social networks (Fan and Shelton, 2009), changes in socio-economic status, and genetic sequence evolution. To realistically model such processes, we need to reason about systems that are composed of multiple components (e.g., many servers in a server farm, multiple residues in a protein sequence) and evolve in continuous time. Continuous-time Bayesian networks (CTBNs) provide a representation language for such processes, which allows to naturally exploit sparse patterns of interactions to compactly represent the dynamics of such processes (Nodelman et al., 2002).

Inference in multi-component temporal models is a notoriously hard problem (Boyen and Koller, 1998). Similar to the situation in discrete time processes, inference is exponential in the number of components, even in a CTBN with sparse interactions (Nodelman et al., 2002). Thus, we have to resort to approximate inference methods. The recent literature has adapted several strategies from discrete graphical models to CTBNs. These include sampling-based approaches, where Fan and Shelton (2008) introduce a likelihood-weighted sampling scheme, and more recently El-Hay et al. (2008) introduce a Gibbs-sampling procedure. Such sampling-based approaches yield more accurate answers with the investment of additional computation. However, it is hard to bound the required time in advance, tune the stopping criteria, or estimate the error of the approximation. An alternative class of approximations is based on *variational principles*.

Recently, Nodelman et al. (2005b) and Saria et al. (2007) introduced an *Expectation Propagation* approach, which can be roughly described as a local message passing scheme, where each message describes the dynamics of a single component over an interval. This message passing procedure can be efficient. Moreover it can automatically refine the number of intervals according to the complexity of the underlying system. Nonetheless, it does suffer from several caveats. On the formal level, the approximation has no convergence guaranties. Second, upon convergence, the computed marginals do not necessarily form a globally consistent distribution. Third, it is restricted to approximations in the form of piecewise-homogeneous messages on each interval. Thus, the refinement of the number of intervals depends on the fit of such homogeneous approximations to the target process. Finally, the approximation of Nodelman *et al* does not provide a provable approximation on the likelihood of the observation—a crucial component in learning procedures.

1.2 Our Contribution

Here, we develop an alternative variational approximation, which provides a different tradeoff. We use the strategy of structured variational approximations in graphical models (Jordan et al., 1998), and specifically by the variational approach of Opper and Sanguinetti (2007) for approximate inference in Markov Jump Processes, a related class of models (see below for more elaborate comparison). The resulting procedure approximates the posterior distribution of the CTBN as a product of independent components, each of which is a inhomogeneous continuous-time Markov process. As we show, by using a natural representation of these processes, we derive a variational procedure that is both efficient, and provides a good approximation both for the likelihood of the evidence and for the expected sufficient statistics. In particular, the approximation provides a lower-bound on the likelihood, and thus is attractive for use in learning.

Chapter 2

Foundations

In this chapter we will discuss the foundations required for understanding our work on multi component continuous-time Markov processes. The natural representation for these processes, due to their inherent structure, are with *graphical models*. First, we will discuss general joint probabilities. Next we will move to discrete Markov processes, and finally we will present continuous-time Markov processes. For each of these models, we will discuss their representation and inference.

Graphical models are an elegant framework which uses the structure of a distribution to compactly represent the joint distribution of a set of random variables. A notational convention: vectors are denoted by boldface symbols, e.g., $\boldsymbol{X}$, with state space $S = S_1 \times S_2 \times \cdots \times S_D$. The states in S are denoted by vectors of indexes, $\boldsymbol{x} = (x_1, \ldots, x_D)$. We use indexes $1 \leq i, j \leq D$ for enumerating components. Matrices are denoted by blackboard style characters, e.g., $\mathbb{Q}$.

In general, given a set $\boldsymbol{X} = \{X_1, \ldots, X_n\}$, their joint probability distribution requires the specification of a probability $\Pr(x_1, \ldots, x_n)$ for every possible assignment. This results in an exponential blowup with respect to the number of variables in the set. However, using the dependence structure between the different variables allows graphical models to factor the representation of the distribution into modular components. This structure exists in many real-world phenomena, and is what makes this framework so appealing. Additionally, effective learning in probablistic graphical models is enabled by the compact representation.

2.1 Bayesian Networks

One of the most common classes of graphical models are *Bayesian networks*, whos underlying semantics are based on directed graphs, hence they are also called *directed graphical models*. These models allow compact representation of the joint distribution of a set of random variables, using the independencies in the probability space.

2.1.1 Representation

The core of the Bayesian network is a directed acyclic graph (DAG). Each vertex corresponds to a random variable and the topology of the graph relates to the dependency structure of the

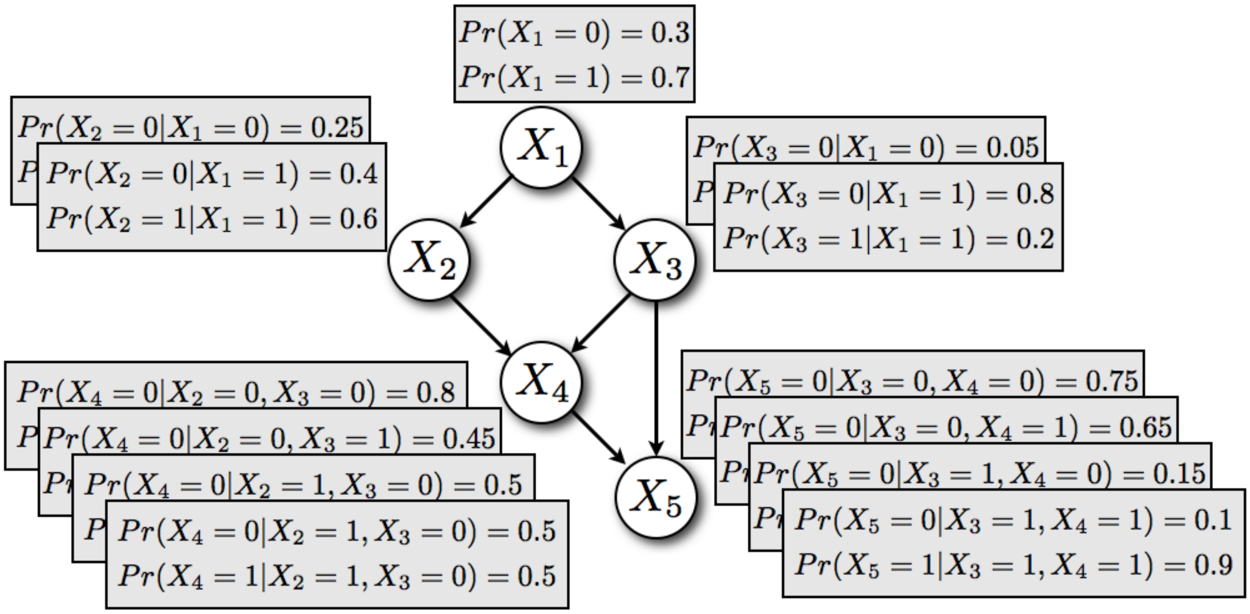
$Pr(X_1 = 0) = 0.3$
$Pr(X_1 = 1) = 0.7$
$Pr(X_2 = 0|X_1 = 0) = 0.25$
$Pr(X_2 = 0|X_1 = 1) = 0.4$
$Pr(X_2 = 1|X_1 = 1) = 0.6$
$Pr(X_3 = 0|X_1 = 0) = 0.05$
$Pr(X_3 = 0|X_1 = 1) = 0.8$
$Pr(X_3 = 1|X_1 = 1) = 0.2$
X_1
X_2
X_3
X_4
X_5
$Pr(X_4 = 0|X_2 = 0, X_3 = 0) = 0.8$
$Pr(X_4 = 0|X_2 = 0, X_3 = 1) = 0.45$
$Pr(X_4 = 0|X_2 = 1, X_3 = 0) = 0.5$
$Pr(X_4 = 0|X_2 = 1, X_3 = 0) = 0.5$
$Pr(X_4 = 1|X_2 = 1, X_3 = 0) = 0.5$
$Pr(X_5 = 0|X_3 = 0, X_4 = 0) = 0.75$
$Pr(X_5 = 0|X_3 = 0, X_4 = 1) = 0.65$
$Pr(X_5 = 0|X_3 = 1, X_4 = 0) = 0.15$
$Pr(X_5 = 0|X_3 = 1, X_4 = 1) = 0.1$
$Pr(X_5 = 1|X_3 = 1, X_4 = 1) = 0.9$

Figure 2.1: A simple Bayesian network consisting of 5 binary components. Each component's CPDs are shown, one for each possible state of its parents.

underlying distribution. Each vertex X_i is associated with a conditional probability distribution (CPD) $Pr(X_i = x_i|\mathbf{Pa}_i = \boldsymbol{u}_i)$ specifying the conditional probability of the variable being in each possible state x_i conditioned on the state $\boldsymbol{u}_i$ of its parents $\mathbf{Pa}_i$. There are numerous ways to represent this conditional distribution. The simplest, albeit not the most efficient, method is using a table that stores the conditional probability of each state of the component for each possible assignment of its parents. These are often referred to as *table CPDs*.

Example 2.1.1: For example, let us look at the network depicted in Fig. 2.1. Here we have a simple 5 component network, where each component has 2 possible states (0 ot 1). The CPDs of each component specify the probability of being in each possible state conditioned on each of the states of its parents. For example, X_1 requires only 1, because it is a root. On the other hand, representing the factorized distribution of X_5 requires 4 tables, one for every possible state of X_4 and X_3. ▮

Definition 2.1.2: A probability distribution P *factorizes over* $\mathcal{G}$ if for every assignment $\boldsymbol{x} = \{x_1, \ldots, x_n\}$ of $\mathcal{X}$ the joint probability can be factorized according to the independence structure of the graph:

$$Pr(\boldsymbol{X} = \boldsymbol{x}) = \prod_{i=1}^{D} Pr(X_i = x_i|\mathbf{Pa}_i = \boldsymbol{u}_i) \tag{2.1}$$

Example 2.1.3: Returning to example Example 2.1.1, the probability distribution factorizes according to (2.1) as

$$\begin{aligned}&\Pr(\boldsymbol{X} = (x_1, x_2, x_3, x_4, x_5)) = \\ &\qquad \Pr(X_1 = x_1) \Pr(X_2 = x_2|x_1) \Pr(X_3 = x_3|x_1) \Pr(X_4 = x_4|x_2, x_3) \Pr(X_5 = x_5|x_3, x_4) \ .\end{aligned}$$

This exploitation of structure avoids the need to enumerate over all possible assignments, leading to an efficient calculation. ▮

2.1.2 Inference in Bayesian Networks

One of the purposes of a probabilistic model is to compactly represent the joint distribution of the set of random variables. This joint distribution and its structure hold certain properties we may wish to query about for different reasons. The calculation of these queries is called *inference* and is generally a difficult task.

In some cases, we would like to reason about networks where the states of some of the components are known. We refer to these observed states as *evidence* which we simply denote as $\boldsymbol{e}$. An important inference query is the calculation of the *likelihood of the evidence*. For example, when we observe the state of the components $\boldsymbol{e} = \{X_2 = x_2, X_3 = x_3\}$, we would like to calculate the probability of this assignment. To find this probability, we enumerate over all the states of the remaining undetermined components

$$\Pr(\boldsymbol{e}) = \sum_{x_1, x_4, x_5} \Pr(X_1 = x_1, X_2 = x_2, X_3 = x_3, X_4 = x_4, X_5 = x_5) \ .$$

Naturally, the probability of an assignment that is not consistent with the evidence is 0, thus we only enumerate over assignments of the remaining variables, fixing the values of the observed ones.

The next set of queries is the calculation of the *posterior probability*. As opposed to the *prior probablity* which implies that we are reasoning before seeing any evidence, the posterior is the probability distribution $\Pr(\boldsymbol{U}|\boldsymbol{e})$, whose sample space is all possible states of some subset of the components $\boldsymbol{U} \subseteq \boldsymbol{X}$.

Generally, this may be a difficult task, because it would require summing out all of the other components by enumeration over all their different states. To show this we use the *Law of total probability*, and write the probability

$$\Pr(\boldsymbol{u}|\boldsymbol{e}) = \sum_{\boldsymbol{x} \setminus \{\boldsymbol{e}, \boldsymbol{u}\}} \Pr(\boldsymbol{x}, \boldsymbol{u}|\boldsymbol{e})$$

Let us return to Example 2.1.1. Here we may wish to calculate the marginal probability that $X_2 = x_2$ and $X_3 = x_3$, given that $X_3 = x_3$, i.e., - $\Pr(\boldsymbol{X}_2 = x_2, X_5 = x_5|X_3 = x_3)$. This can be performed naively by calculating

$$\begin{aligned}\Pr(X_2 = x_2, X_5 = x_5 | X_3 = x_3) &= \sum_{x_1,x_4} \Pr(x_1, x_2, x_4, x_5 | x_3) \\ &= \frac{\sum_{x_1,x_4} \Pr(x_1, x_2, x_3, x_4, x_5)}{\sum_{x_1,x_2,x_4,x_5} \Pr(x_1, x_2, x_3, x_4, x_5)} \ .\end{aligned}$$

In the general case, this summing out operation is exponential in the number of components, which makes inference in arbitrary networks NP-hard and so is intractable. Later we shall see how in some cases we can exploit the network structure to perform this operation more efficiently.

The final Bayesian network inference query aims to find, given evidence, the *maximum a-posteriori* (*MAP*) assignment of the remaining variables $\boldsymbol{U} = \boldsymbol{X} \setminus \boldsymbol{e}$. Basically, the MAP query aims to compute

$$\arg\max_{\boldsymbol{u}} \Pr(\boldsymbol{u}|\boldsymbol{e}) \ .$$

MAP queries are used in many applications, for example in protein design (Fromer and Yanover, 2009). Even though this query is supposedly easier than the marginal queries, it is still NP-hard. Next, we will explore the different methods for calculating these values.

Variable Elimination

In some cases, the summing out procedure shown in the previous section can be simplified if the joint distribution has a *tree structure*, i.e., its corresponding Bayesian network is a tree. Specifically, if the graph of the Bayesian network is a tree, we can sum out the other components in linear time where each step we “eliminate” a different variable (Pearl, 1988).

We can also use the variable elimination algorithm to solve the MAP queries by exchanging the sums with max. Even though this operation is efficient and exact in tree graphs, arbitrary networks are rarely without cycles. Therefore, we must resort to approximations.

Approximate Inference

In this section we will see some approximation methods for inference in general graphs. In all of them we will approximate the required expectations using various methods, each of which has different trade-offs.

Sampling The first group of approximations are *particle based approximations*. As their names suggests, these algorithms calculate the inference queries by averaging over samples (otherwise named *particles*). The justification for this approach comes from the *law of large numbers* which states that any finite integral of a function can be estimated by sampling from that function with probability proportional to its value. More generally, the expectation of any random variable X with mean μ_X can be estimated by sampling it repeatedly and setting

$$\mu_X = \frac{1}{N} \sum_{n=1}^{N} s_n$$

```
$\boldsymbol{S} = \phi$
for $i$ = 1 to $N$ do
    randomly initialize $\boldsymbol{S}^i = \boldsymbol{X}^0$, $j = 0$
    while not mixed do
        pick random component $k \in \{1, \ldots, D\}$
        sample $\boldsymbol{S}^i_k$ given all components $\neq k$ in $\boldsymbol{S}^i$
    end
    insert $\boldsymbol{S} \leftarrow \boldsymbol{S}^i$
end
```

Figure 2.2: Abstract Gibbs sampling with dependencies

where s_n is the n^{th} sample and N is the number of samples. Additionally, the error of the approximation declines exponentially with the number of samples taken. Sampling from a distribution with dependencies is not a trivial task, and requires incorporating the evidence into the distribution we sample from. Specifically, for the mean of the samples to be close to the real mean we must take a number of *i.i.d* samples, which means they are independent and identically distributed. Just sampling arbitrarily will not meet these requirements.

Markov Chain Monte Carlo algorithms allow us to sample from distributions with certain properties, and in particular, those with dependencies. The *Gibbs sampling* procedure is one that correctly samples from the distribution we want to approximate by producing a chain of samples, each sampled conditioned on the previous one. When the chain is long enough we can assume that the sample we get is independent of its initialization. The algorithm can be abstractly viewed in 2.2, where we sample N independent samples.

To get the i^{th} sample correctly, we start at some arbitrary global state $\boldsymbol{X}^0$. In each iteration, we first sample the component to be changed k and then we sample its new state given the states of all the others from the conditional distribution

$$\begin{aligned}&\Pr(X_k = x_k | X_1 = x_1, \ldots, X_D = x_D) \\ &= \frac{\Pr(X_1 = x_1, \ldots, X_k = x_k, \ldots, X_D = x_d)}{\sum_{y_k} \Pr(X_1 = x_1, \ldots, X_k = y_k, \ldots, X_D = x_d)} .\end{aligned}$$

After updating the k^{th} component, we simply choose another one at random and run the same update rule as before. Depending on different properties of the system such as its *mixing time*, which determines how long a chain is required for the sampling to be correct. After performing enough iterations our the sample, which we denote $\boldsymbol{S}^1$, is sampled from the target distribution. Now we can perform this procedure again and again to obtain a sequence of N independent samples $\boldsymbol{S}^1, \ldots, \boldsymbol{S}^N$.

Sampling algorithms are strong inference tools, as they hold the *any-time property*, which means they improve their estimate as they run longer. Additionally, they are asymptotically unbiased, which means they are guarantied to converge to the exact answer if run long enough. On the down-side, it is very difficult to know when they have converged, and also to estimate their error. Sampling algorithms may also require many iterations to converge, depending on the system's mixing time and the number of components, among others.

Message passing The message passing paradigm is a general framework in which there are numerous algorithms (Talya Meltzer and Weiss, 2009). These algorithms always converge and give the exact marginal distributions on trees. Even though they do not have any guaranties on graphs with cycles, empirically they work surprisingly well (Murphy et al., 1999).

These algorithms have a simple and elegant scheme, yet there is no obvious reason why it should converge to any meaningful result. In general graphs with cycles, there are no convergence guaranties and if the algorithm does converge the marginals it calculates, even though locally consistent (there is a local agreement between neighbouring components) the resulting marginals may not come from a valid distribution (Yedidia et al., 2005).

The next group of algorithms are those which always converge to a consistent distribution. They will also approximate the log-likelihood of the evidence and allow tractable inference.

2.1.3 Mean Field Approximations in Bayesian Networks

A different approach, generally named *Variational approximations* to the abovementioned ones are to cast the inference problem into an optimization of a *functional* (Jordan et al. (1998), Wainwright and Jordan (2008)). Generally, a functional is a mapping from a linear space to the underlying field (usually the reals $\mathbb{R}$). Formally,

Definition 2.1.4: Let $\mathcal{X}$ be a linear space. A functional on $\mathcal{X}$ is a mapping $\mathcal{X} \to \mathbb{R}$, i.e., a real valued function. ▮

Defining this functional in a certain way will see to it that reaching a fixed point in its optimization will give us the best approximation on the marginals. For certain formulations of this functional, for example in the *Mean Field* approximation, this functional may also give us an approximation of the log-likelihood of the evidence. This functional represents the "distance" between the original intractable model and an approximate model, which is tractable and will allow us to approximate the marginals. In the next section we will unfold the transition of the inference problem into an optimization scheme.

Variational Principle

In the presence of observations $\boldsymbol{e}$, we can write the *posterior distribution*

$$\mathbf{P}(\boldsymbol{x}|\boldsymbol{e}) = \frac{\mathbf{P}(\boldsymbol{x},\boldsymbol{e})}{\mathbf{P}(\boldsymbol{e})} = \frac{\mathbf{P}(\boldsymbol{x},\boldsymbol{e})}{Z}$$

where Z is the *partition function*

$$Z = \sum_{\boldsymbol{x}\setminus\boldsymbol{e}} \mathbf{P}(\boldsymbol{x},\boldsymbol{e}) \ .$$

The calculation of the partition function requires enumeration over all possible assignments and so is usually intractable. In the case of Bayesian networks this term equals the log-likelihood of the evidence. For some arbitrary distribution $\mathbf{Q}(x)$ we define the *free energy* between $\mathbf{Q}$ and the posterior

$$\mathcal{F}(\mathbf{Q}; \mathbf{P}(\cdot|\boldsymbol{e})) = \mathcal{H}(\mathbf{Q}) + \mathcal{E}(\mathbf{Q}, \mathbf{P}(\cdot|\boldsymbol{e})) \tag{2.2}$$

where the first term is the *entropy* defined as

$$\mathcal{H}(\mathbf{Q}) = -\mathbf{E}_{\mathbf{Q}}\left[\ln \mathbf{Q}\right]$$

and the second is the *average energy*

$$\mathcal{E}(\mathbf{Q}(\cdot), \mathbf{P}(\cdot|\boldsymbol{e})) = \mathbf{E}_{\mathbf{Q}}\left[\ln \mathbf{P}(\boldsymbol{x}, \boldsymbol{e})\right] \ .$$

One standard way of measuring the "distance" between distributions is the *KL divergence*. This measure between two arbitrary distributions is defined as

$$\mathcal{D}(\mathbf{Q}\|\mathbf{P}(\cdot|\boldsymbol{e})) = \mathbf{E}_{\mathbf{Q}}\left[\ln \frac{\mathbf{Q}}{\mathbf{P}(\cdot|\boldsymbol{e})}\right] = \sum_{\boldsymbol{x}} \mathbf{Q}(x) \ln \frac{\mathbf{Q}(\boldsymbol{x})}{\mathbf{P}(\boldsymbol{x}|\boldsymbol{e})} \ . \tag{2.3}$$

This divergence, although not a full *metric*, does have some useful properties when both $\mathbf{P}(\cdot|\boldsymbol{e})$ and $\mathbf{Q}$ are distributions. It is always non-negative, convex and is equal to zero iff $\mathbf{Q} = \mathbf{P}(\cdot|\boldsymbol{e})$. Additionally it gives a good measure to the distance between the two distributions.

In our case we want to approximate the posterior distribution $\mathbf{P}(\cdot|\boldsymbol{e})$. The KL-divergence from the approximation $\mathbf{Q}$ is thus

$$\begin{aligned} \mathcal{D}(\mathbf{Q}\|\mathbf{P}(\cdot|\boldsymbol{e})) &= \mathbf{E}_{\mathbf{Q}}\left[\ln \frac{\mathbf{Q}}{\mathbf{P}(\cdot|\boldsymbol{e})}\right] \\ &= \mathbf{E}_{\mathbf{Q}}\left[\ln \mathbf{Q}\right] - \mathbf{E}_{\mathbf{Q}}\left[\ln \mathbf{P}(\cdot, \boldsymbol{e})\right] + \ln \mathbf{P}(\boldsymbol{e}) \end{aligned}$$

where the second equality is given by $\mathbf{P}(\cdot|\boldsymbol{e}) = \frac{\mathbf{P}(\cdot,\boldsymbol{e})}{\mathbf{P}(\boldsymbol{e})}$ and the fact that $\mathbf{P}(\boldsymbol{e})$ is independent of $\mathbf{Q}$. Defining the free energy as in (2.2) we get that

$$\mathcal{F}(\mathbf{Q}; \mathbf{P}(\cdot|\boldsymbol{e})) = \ln \mathbf{P}(\boldsymbol{e}) - \mathcal{D}(\mathbf{Q}\|\mathbf{P}(\cdot|\boldsymbol{e})) \ .$$

This equation has two important implications for our approximation. The first is that maximizing the functional is equivalent to minimizing the KL-divergence. If we can write the functional in a tractable form it will be possible to perform optimization over it and even calculate its value. Secondly, because the KL-divergence is non-negative, we can see that as long as $\mathbf{Q}$ is a consistent probability the functional is a lower bound on the log-likelihood of the evidence. This fact is crucial for learning task, which we will not discuss here.

If we allow the approximation $\mathbf{Q}$ to take any form, then obviously the best approximation for $\mathbf{P}$ is itself. On the one hand, setting $\mathbf{Q}(\boldsymbol{x}) = \mathbf{P}(\boldsymbol{x}|\boldsymbol{e})$ gives us the maximal functional value of $\ln \mathbf{P}(\boldsymbol{e})$, yet by doing so all we have obtained is another intractable distribution which will not ease our calculations. Thus, our aim is to approximate $\mathbf{P}(\cdot|\boldsymbol{e})$ with a *tractable* distribution $\mathbf{Q}$, and more specifically the "best" possible one taken from some family $\mathcal{M}$ of tractable distributions. We now define our optimization

Find $\mathbf{Q}^* = \arg\max_{\mathbf{Q}\in\mathcal{M}} \mathcal{F}(\mathbf{Q}; \mathbf{P}(\cdot|\boldsymbol{e}))$

The choice of the family $\mathcal{M}$ is non-trivial and should be problem specific. The *Mean Field* approach approximates $\mathbf{P}(\cdot|\boldsymbol{e})$ with fully factored probabilities. This approach, after the optimization phase, yields the closest factorized probability distribution $\mathbf{Q}^*$. This may seem like a crude approximation to a possibly complex and informative target distribution, but in many problems gives satisfactory results. Another advantage of this approach is that the resulting $\mathbf{Q}^*$ is simple to work with, and the calculation of the marginals and statistics becomes linear in the number of components (as we perform inference on each component separately). More generally, we can define *structured Mean Field* approximations, where we retain some structure in the approximating distribution. This in turn may give us a much closer approximation, but on the other hand make the inference on the result more difficult.

In order to have a better understanding of these variational methods, we shall delve into the Mean Field approximation in Bayesian networks.

Mean Field

The standard Mean Field (Koller and Friedman, 2009) approximation attempts to approximate the original distribution by one that assumes independence between all the variables. Namely, we would like to approximate $\mathbf{P}(\cdot|\boldsymbol{e})$ using distributions $\mathbf{Q}(\boldsymbol{X})$ constrained to the family of factorized distributions

$$\mathcal{M}_{fact} = \left\{ \mathbf{Q}(\boldsymbol{X}) : \mathbf{Q}(\boldsymbol{X}) = \prod_{i=1}^{D} \mathbf{Q}(X_i) \right\}$$

where we denote the approximate distribution over the subset of components $\boldsymbol{U}$ as $\mathbf{Q}(\boldsymbol{U})$, and particularly over the i^{th} component as $\mathbf{Q}(X_i)$. This is equivalent to removing all the edges in the Bayesian network which represents the original distribution. Our optimization problem is to find $\mathbf{Q}$ that is closest to our original distribution, formally

Find $\mathbf{Q}^* = \arg\min_{\mathbf{Q}} \mathcal{D}(\mathbf{Q}\|\mathbf{P}(\cdot|\boldsymbol{e}))$
s.t. $\mathbf{Q} \in \mathcal{M}_{fact}$

The Mean Field algorithm is derived by considering the fixed points of the energy functional. This functional will have a much simpler form due to the independence of the approximating distribution. The entropy of distributions from $\mathcal{M}_{fact}$ can be written as the sum of local entropies

$$\mathcal{H}(\mathbf{Q}(\boldsymbol{X})) = \sum_{i=1}^{D} \mathcal{H}(\mathbf{Q}(X_i))$$

Proof:

$$\begin{aligned}\mathcal{H}(\mathbf{Q}(\boldsymbol{X})) &= \sum_{\boldsymbol{x}} \mathbf{Q}(\boldsymbol{x}) \ln \mathbf{Q}(\boldsymbol{x}) \\ &= \sum_{\boldsymbol{x}} \prod_{j} \mathbf{Q}(x_j) \ln \prod_{i} \mathbf{Q}(x_i) \\ &= \sum_{i} \sum_{\boldsymbol{x}} \prod_{j} \mathbf{Q}(x_j) \ln \mathbf{Q}(x_i)\end{aligned}$$

We can enumerate over all $\boldsymbol{x}$ using a specific order

$$\begin{aligned}\sum_{i} \sum_{\boldsymbol{x}} \prod_{j} \mathbf{Q}(x_j) \ln \mathbf{Q}(x_i) &= \sum_{i} \sum_{x_i} \mathbf{Q}(x_i) \ln \mathbf{Q}(x_i) \sum_{\boldsymbol{x}\backslash i} \prod_{j\backslash i} \mathbf{Q}(x_j) \\ &= \sum_{i} \sum_{x_i} \mathbf{Q}(x_i) \ln \mathbf{Q}(x_i)\end{aligned}$$

where the last equality is true because

$$\sum_{\boldsymbol{x}\backslash i} \prod_{j\backslash i} \mathbf{Q}(x_j) = \sum_{\boldsymbol{x}\backslash i} \mathbf{Q}(\boldsymbol{X}) = 1 \ . \tag{2.4}$$

This brings us to the conclusion that

$$\mathcal{H}(\mathbf{Q}(\boldsymbol{X})) = \sum_{i} \sum_{x_i} \mathbf{Q}(x_i) \ln \mathbf{Q}(x_i) = \sum_{i} \mathcal{H}(\mathbf{Q}(X_i))$$

∎

Additionally, we can write the average free energy as a sum of local terms

$$\mathcal{E}(\mathbf{Q}, \mathbf{P}(\cdot|\boldsymbol{e})) = \sum_{i=1}^{D} \sum_{x_i, \boldsymbol{u}_i} \mathbf{Q}(x_i, \boldsymbol{u}_i) \ln \mathbf{P}(x_i|\boldsymbol{u}_i)$$

Proof: Using the definition of the average free energy

$$\mathcal{E}(\mathbf{Q}, \mathbf{P}(\cdot|\boldsymbol{e})) = \sum_{\boldsymbol{x}} \mathbf{Q}(\boldsymbol{x}) \ln \mathbf{P}(\boldsymbol{x}|\boldsymbol{e}) \ .$$

Given the factorized structure of the distributions we can write the previous term

$$\begin{aligned}\sum_{\boldsymbol{x}\backslash \boldsymbol{e}} \mathbf{Q}(\boldsymbol{x}) \ln \mathbf{P}(\boldsymbol{x}|\boldsymbol{e}) &= \sum_{i} \sum_{\boldsymbol{x}\backslash \boldsymbol{e}} \mathbf{Q}(\boldsymbol{x}) \ln \mathbf{P}(x_i|\boldsymbol{u}_i) \\ &= \sum_{i} \sum_{\boldsymbol{x}\backslash \boldsymbol{e}} \prod_{j} \mathbf{Q}(x_j), \ln \mathbf{P}(x_i|\boldsymbol{u}_i) \ .\end{aligned}$$

Now we can change the order of summation and get

$$\begin{aligned}\sum_i \sum_{\boldsymbol{x}\setminus\boldsymbol{e}} \prod_j \mathbf{Q}(x_j), \ln \mathbf{P}(x_i|\boldsymbol{u}_i) &= \sum_i \sum_{x_i,\boldsymbol{u}_i} \mathbf{Q}(x_i,\boldsymbol{u}_i) \ln \mathbf{P}(x_i|\boldsymbol{u}_i) \sum_{\boldsymbol{x}\setminus\boldsymbol{e}\cup\{i,\mathbf{Pa}_i\}} \prod_{j\in\boldsymbol{x}\setminus\boldsymbol{e}\cup\{i,\mathbf{Pa}_i\}} \mathbf{Q}(x_j) \\ &= \sum_i \sum_{x_i,\boldsymbol{u}_i} \mathbf{Q}(x_i,\boldsymbol{u}_i) \ln \mathbf{P}(x_i|\boldsymbol{u}_i)\end{aligned}$$

where the last equality follows from a similar principle to (2.4). ▮

Here we see that the complexity of this term depends solely on the structure of $\mathbf{P}(\boldsymbol{X})$, and can be calculated in a complexity that is exponential in the largest set of parents of any component.

These previous proofs give us a new factorized functional

$$\tilde{\mathcal{F}}(\mathbf{Q};\mathbf{P}(\cdot|\boldsymbol{e})) = \sum_i \sum_{x_i} \mathbf{Q}(x_i) \ln \mathbf{Q}(x_i) + \sum_{i=1}^{D} \sum_{x_i,\boldsymbol{u}_i} \mathbf{Q}(x_i,\boldsymbol{u}_i) \ln \mathbf{P}(x_i|\boldsymbol{u}_i) \ . \tag{2.5}$$

Now we have the following optimization problem:

Find $\mathbf{Q}(\boldsymbol{x}) = \prod_{i=1}^{D} \mathbf{Q}(x_i)$

that maximizes $\arg\max_{\mathbf{Q}} \tilde{\mathcal{F}}(\mathbf{Q};\mathbf{P}(\cdot|\boldsymbol{e}))$

s.t. $\sum_{x_i} \mathbf{Q}(x_i) = 1 \quad \forall\, i$

Here we have a *constrained optimization* problem, where we are faced with optimizing a functional, under certain constraints. In our case we need the $\mathbf{Q}$ of each component to remain a distribution. The reason we do not enforce their non-negativity will be clear when we see the way we update their values. Using the Lagrange multiplier theory (Appendix A.1), we will perform constrained optimization over our functional in (2.5).

Corollary 2.1.5: *The stationary point of the energy functional $\tilde{\mathcal{F}}(\mathbf{Q};\mathbf{P}(\cdot|\boldsymbol{e}))$ with respect to $\mathbf{Q}(x_i)$ is*

$$\frac{1}{Z_i} \exp\left\{-\boldsymbol{E}_{\mathbf{Q}}[\ln \mathbf{P}(\cdot|\boldsymbol{e}) \mid x_i]\right\}$$

where

$$Z_i = \sum_{x_i} \exp\left\{\boldsymbol{E}_{\mathbf{Q}}[\ln \mathbf{P}(\cdot|\boldsymbol{e}) \mid x_i]\right\}$$

Proof: We define the Lagrangian as

$$\mathcal{L} = \tilde{\mathcal{F}}(\mathbf{Q};\mathbf{P}(\cdot|\boldsymbol{e})) - \sum_i \lambda_i \left(\sum_{x_i} \mathbf{Q}(x_i) - 1 \right)$$

where $\tilde{\mathcal{F}}(\mathbf{Q};\mathbf{P}(\cdot|\boldsymbol{e}))$ is defined as in (2.5) and λ_i are the *Lagrange multipliers* that enforce the constraints on the marginals of $\mathbf{Q}$.

We can further write the parts of the Lagrangian that concern the marginal $\mathbf{Q}(x_i)$ and over which we optimize this parameter

$$\mathcal{L}_i = \tilde{\mathcal{F}}(\mathbf{Q}; \mathbf{P}(\cdot|\boldsymbol{e})) - \lambda_i \left(\sum_{x_i} \mathbf{Q}(x_i) - 1 \right) \ .$$

Notice that this is a convex function of $\mathbf{Q}(x_i)$. Calculating the partial derivatives of $\mathcal{L}_i$ gives us

$$\frac{\partial}{\partial \mathbf{Q}(x_i)} \mathcal{L}_i \ = \ -\ln \mathbf{Q}(x_i) + \mathbf{Q}(\boldsymbol{u}_i) \ln \mathbf{P}(x_i|\boldsymbol{u}_i) + \sum_{j \in \mathit{Children}_i} \sum_{\boldsymbol{u}_j \setminus i} \mathbf{Q}(x_j, \boldsymbol{u}_j) \ln \mathbf{P}(x_j|\boldsymbol{u}_j, x_i) - \lambda^i \ .$$

Setting the derivatives to zero and rearranging terms we obtain the following equation

$$\ln \mathbf{Q}(x_i) = -\mathbf{Q}(\boldsymbol{u}_i) \ln \mathbf{P}(x_i|\boldsymbol{u}_i) - \sum_{j \in \mathit{Children}_i} \sum_{\boldsymbol{u}_j \setminus i} \mathbf{Q}(x_j, \boldsymbol{u}_j) \ln \mathbf{Q}(x_j|\boldsymbol{u}_j, x_i) + \lambda^i \ .$$

Taking exponents on both sides, we can calculate the new parameterization given the current marginals of all the other components and the terms and the parameters of $\mathbf{P}$, and then normalize all the marginals to obtain the constraint that $\mathbf{Q}$ is a distribution over the i^{th} component. The right hand side of the previous equation is exactly the expectation we require, as

$$\mathbf{Q}(\boldsymbol{u}_i) \ln \mathbf{P}(x_i|\boldsymbol{u}_i) + \sum_{j \in \mathit{Children}_i} \sum_{\boldsymbol{u}_j \setminus i} \mathbf{Q}(x_j, \boldsymbol{u}_j) \ln \mathbf{Q}(x_j|\boldsymbol{u}_j, x_i) = \boldsymbol{E}_{\mathbf{Q}}[\ln \mathbf{P}(\cdot|\boldsymbol{e}) \mid x_i] \ .$$

Finally, because λ_i is the same for all x_i it will vanish with this normalization and so can be discarded.

The final touch of the optimization of the i^{th} component comes from the derivative of the lagrangian w.r.t. λ_i, which forces us to normalize the marginals $\mathbf{Q}(x_i)$. When updating the i^{th} component we calculate the normalizing constant of this component as

$$Z_i = \sum_{x_i} \exp \left\{ \boldsymbol{E}_{\mathbf{Q}}[\ln \mathbf{P}(\cdot|\boldsymbol{e}) \mid x_i] \right\}$$

and update each component

$$\mathbf{Q}(x_i) = \frac{1}{Z_i} \exp \left\{ \boldsymbol{E}_{\mathbf{Q}}[\ln \mathbf{P}(\cdot|\boldsymbol{e}) \mid x_i] \right\} \ . \tag{2.6}$$

The unnormalized update equations of the different marginals are independent of each other, as they depend only on the marginals of the other components, and are invariant to multiplication by a normalizing constant. Therefore, after this normalization step the marginals still form a fixed point of the functional, and now also satisfy the normalization constraints. ∎

Now we can use Corollary 2.1.5 to get the update equations for the optimization. The algorithm, which is shown in 2.3, iteratively chooses a random component and updates its current marginals using (2.6).

```
for each $i = 1, \ldots, D$
    initialize $\mathbf{Q}(x_i) = \mathbf{Q}(x_i | \boldsymbol{u}_i)$ for some random state $\boldsymbol{u}_i$
while not converged do
    1. pick random component $i \in \{1, \ldots, D\}$
    2. update marginals using (2.6)
end
Output: marginals $\mathbf{Q}(x_i)$ for all $i = 1, \ldots, D$
```

Figure 2.3: Mean Field approximation in Bayesian networks

Notice that the update of the i^{th} component does not depend on our current belief on the values of the marginals of that component, which means the fixed point equations of each component are self-consistent. This, along with the fact that $\mathcal{L}_i$ is convex in $\mathbf{Q}(x_i)$ gives us the guaranty that at each update of the parameters of the i^{th} component we reach the coordinate-wise optimum, thus increasing the functional in each iteration. This functional is upper bounded by the log-likelihood of the evidence, and so the procedure will ultimately converge to a local optimum.

2.2 Discrete-Time Markov Processes

Many dynamic systems can be represented as discrete stochastic processes (Dean and Kanazawa, 1989) such as the *random walk* (Feller, 1968). A stochastic process $X^{(t)}$ involves the dynamics of a stochastic system in time, where $X^{(t)}$ is a random variable which denotes its state at time t.

The system can be in a number of different states, which can change in regular intervals $t_0, t_1, \ldots$ with some probability $\Pr(X^{(t_{k+1})} = x' | X^{(t_k)} = x)$. We are interested particularly *Markov* processes, which satisfy the *Markov* property.

Definition 2.2.1: A process is said to satisfy the *Markov property* (and is thus a *Markov process*) if, for any set of real valued indices $\{t_0 < t_1 < \ldots < t_K\}$, the following independence property is fulfilled:

$$\Pr(X^{(t_K)} | X^{(t_0)} \ldots X^{(t_{K-1})}) = \Pr(X^{(t_K)} | X^{(t_{K-1})}) \ .$$

▮

2.2.1 Representation

In order to represent the process in a succinct manner, we make another assumption on its parameterization

Definition 2.2.2: The dynamics of a *time-homogeneous Markov process* are fully determined by the *Markov transition function*,

$$p_{x,y}(t) = \Pr(X^{(t+s)} = y | X^{(s)} = x),$$

where time-homogeneity implies that the right-hand side does not depend on s. ▮

A time-homogeneous stochastic Markov process can be represented using a *prior distribution* $\pi(\boldsymbol{x}) = \Pr(\boldsymbol{X}^{(t_0)} = \boldsymbol{x})$, which states the distribution of the process starting in each of the different possible states, and the *transition probability* p where $p_{a,b} = \Pr(X^{(t_{k+1})} = b | X^{(t_k)} = a)$ is the transition probability from state a to b in one time interval. Notice that the homogeneity assumption is the reason we can represent all the transition distributions at all the points with a single matrix. The next lemma decomposes the probability of the sequence of observations, here called *trajectories*, using these parameters alone.

Lemma 2.2.3: *The joint probability of the trajectory $x^{(t_0)}, \ldots, x^{(t_K)}$ can be written as the product*

$$\Pr(x^{(t_0)}, \ldots, x^{(t_K)}) = \Pr(x^{(t_0)}) \cdot \prod_{k=0}^{K-1} \Pr(x^{(t_{k+1})} | x^{(t_k)})$$

Proof: Using the *Bayes formula*, we can write the probability of the trajectory as

$$\Pr(x^{(t_0)}, \ldots, x^{(t_K)}) = \Pr(x^{(t_0)}) \cdot \prod_{k=1}^{K} \Pr(x^{(t_k)} | x^{(t_{k-1})}, \ldots, x^{(t_0)})$$

and then by the Markov property assumption, each term in the product reduces to

$$\Pr(x^{(t_k)}|x^{(t_{k-1})},\ldots,x^{(t_0)}) = \Pr(x^{(t_k)}|x^{(t_{k-1})})$$

which gives us the required form. ▮

Using Lemma 2.2.3 we can write the joint distribution of the trajectories of length K as

$$\Pr(X^{(t_0)},\ldots,X^{(t_K)}) = \Pr(X^{(t_0)}) \cdot \prod_{k=1}^{K} \Pr(X^{(t_k)}|X^{(t_{k-1})}) \ .$$

Bayes rule states that we can write this as a quotient

$$\Pr(X^{(t_k)}|X^{(t_{k-1})}) = \frac{\Pr(X^{(t_k)}, X^{(t_{k-1})})}{\Pr(X^{(t_{k-1})})}$$

which gives us the following form of the joint probability

$$\Pr(X^{(t_0)},\ldots,X^{(t_K)}) = \Pr(X^{(t_0)}) \cdot \frac{\prod_{k=0}^{K-1} \Pr(X^{(t_k)}, X^{(t_{k+1})})}{\prod_{k=1}^{K-1} \Pr(X^{(t_k)})} \ .$$

We can see from this decomposition that the *joint probabilities* $\mu_{x,y}[t_k, t_{k+1}] = \Pr(X^{(t_k)} = x, X^{(t_{k+1})} = y)$ and the *marginal distributions* $\mu_x[t_k] = \Pr(X^{(t_k)} = x)$ provide an alternative representation for this process.

As in the Bayesian networks framework, we would like to move to multi component processes, where the state $\boldsymbol{X}^{(t_k)}$ is a vector $(x_1^{(t_k)},\ldots,x_D^{(t_k)})$.

2.2.2 Multicomponent Discrete Markov Process

A multicomponent Markov process describes the evolution of a *random vector* $\boldsymbol{X}^{(t_k)}$ in time. Consider a D-component Markov process $\boldsymbol{X}^{(t_k)} = (X_1^{(t_k)}, X_2^{(t_k)}, \ldots X_D^{(t_k)})$. We use $\boldsymbol{X}^{(t_k)}$ and $X_i^{(t_k)}$ to denote the random variable describing the state of the process and its i'th components at time t_k. This process is described by a *Dynamic Bayesian network* (Dean and Kanazawa, 1989), which represents these random vectors in a graph structure (see Fig. 2.4a). The dependencies between the components create influences between components in different time slices. The entaglement between the different components is what causes difficulties in reasoning about these processes. For example, two components that have no immediate connection become dependent on each other (entangled) after several time slices. This results in difficult inference for these models.

Example 2.2.4: A simple 3 component discrete time process can be seen in Fig. 2.4 (a). Each $X_i^{(t_k)}$ influences both itself and $X^{(t_{k+1})}$ at the next time slice. The posterior distribution is induced by the evidence at both ends of the interval and the original transition probability p. ▮

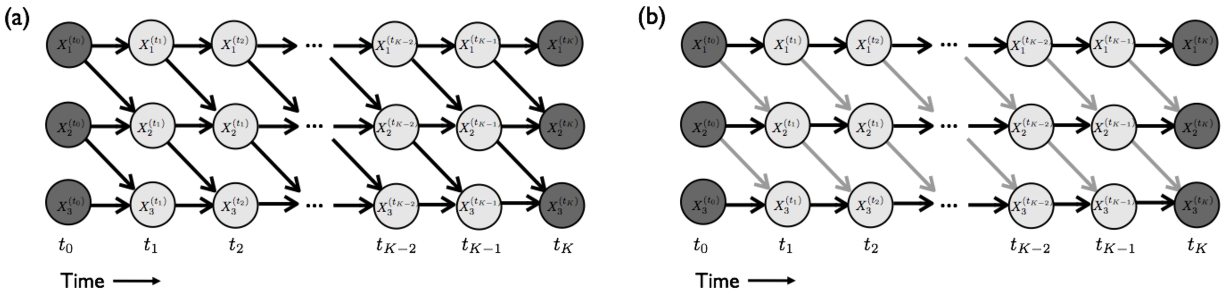
(a)
$X_1^{(t_0)}$ $X_1^{(t_1)}$ $X_1^{(t_2)}$ ⋯ $X_1^{(t_{K-2})}$ $X_1^{(t_{K-1})}$ $X_1^{(t_K)}$
$X_2^{(t_0)}$ $X_2^{(t_1)}$ $X_2^{(t_2)}$ ⋯ $X_2^{(t_{K-2})}$ $X_2^{(t_{K-1})}$ $X_2^{(t_K)}$
$X_3^{(t_0)}$ $X_3^{(t_1)}$ $X_3^{(t_2)}$ ⋯ $X_3^{(t_{K-2})}$ $X_3^{(t_{K-1})}$ $X_3^{(t_K)}$
t_0 t_1 t_2 t_{K-2} t_{K-1} t_K
Time
(b)
$X_1^{(t_0)}$ $X_1^{(t_1)}$ $X_1^{(t_2)}$ ⋯ $X_1^{(t_{K-2})}$ $X_1^{(t_{K-1})}$ $X_1^{(t_K)}$
$X_2^{(t_0)}$ $X_2^{(t_1)}$ $X_2^{(t_1)}$ ⋯ $X_2^{(t_{K-2})}$ $X_2^{(t_{K-1})}$ $X_2^{(t_K)}$
$X_3^{(t_0)}$ $X_3^{(t_1)}$ $X_3^{(t_2)}$ ⋯ $X_3^{(t_{K-2})}$ $X_3^{(t_{K-1})}$ $X_3^{(t_K)}$
t_0 t_1 t_2 t_{K-2} t_{K-1} t_K
Time

Figure 2.4: A dynamic Bayesian network of 3 components with observations on all components at times t_0 and t_K. (a) the original process, with evidence at both ends of the interval. Each $X_i^{(t_k)}$ influences $X_i^{(t_{k+1})}, X_{i+1}^{(t_{k+1})}$. (b) The inhomogeneous Mean Field approximation. Influences only remain between the components and themselves through time.

2.2.3 Inference in Discrete Markov Processes

Inference in discrete processes is similar to those in joint probabilities, as we can represent the process as the evolution of the joint probability distribution over time, like in Fig. 2.4.

As opposed to the Bayesian networks, here the sample space is the set of all trajectories $(X^{(t_0)}, \ldots, X^{(t_K)})$, not single assignments. Given evidence $e = \left\{X^{(t_0)} = e_{t_0}, X^{(t_K)} = e_{t_K}\right\}$ our inference queries are similar to the Bayesian network queries. We will begin by the calculation of *forward probabilities* $\alpha_x(t_k) = \Pr(x^{(t_k)}|e_{t_0})$. These can be calculated using a forward propagation rule in dynamic programming for each $k = 1, \ldots K$

$$\alpha_x(t_k) = \sum_y \alpha_y(t_{k-1}) \cdot p_{y,x}$$

when $\alpha_x(t_0) = \delta_{x,e_{t_0}}$.

Similarly, we can calculate the *backward probabilites* $\beta_x(t_k) = \Pr(e_{t_K}|x^{(k)})$ using the backward propagation rule for all $k = 0, \ldots, K-1$

$$\beta_x(t_k) = \sum_y p_{x,y} \cdot \beta_y(t_{k+1})$$

and $\beta_x(t_K) = \delta_{x,e_{t_K}}$. Next, the likelihood of the evidence $\Pr(e_{t_K}|e_{t_0})$ can be calculated using these values. We can rewrite this likelihood as, for any k

$$\Pr(e_{t_K}|e_{t_0}) = \sum_x \Pr(X^{(t_k)} = x|e_{t_0}) \cdot \Pr(e_{t_k}|X^{(t_k)} = x) = \sum_x \alpha_x(t_k)\beta_x(t_k) \qquad (2.7)$$

which is linear in the number of states, and thus is exponential in the number of components. Alternatively, we can use the fact that the evidence is deterministic, and write

$$\Pr(e) = \alpha_{e_K}(t_K) = \beta_{e_0}(t_0)$$

which would still requires the propagation of either $\beta(t_k)$ backwards or $\alpha(t_k)$ forwards through all the interval. As in any Markov process, we can write any marginal as a function of these three values

$$\begin{aligned}\mu_x[t_k] &= \Pr(X^{(t_k)} = x) \\ &= \frac{\Pr(X^{(t_k)} = x|X^{(t_0)} = e_{t_0}) \cdot \Pr(X^{(t_K)} = e_{t_K}|X^{(t_k)} = x)}{\Pr(X^{(t_K)} = e_{t_K}|X^{(t_0)} = e_{t_0})} \\ &= \frac{\alpha_x(t_k) \cdot \beta_x(t_k)}{\Pr(e)} \ .\end{aligned}$$

Similarly the joint distributions can be written as

$$\mu_{x,y}[t_k, t_{k+1}] = \frac{\alpha_x(t_k) \cdot p_{x,y} \cdot \beta_x(t_{k+1})}{\Pr(e)} \ .$$

Two additional queries are the process' *sufficient statistics*.

Definition 2.2.5: The *sufficient statistics* of a statistical model with a parameter θ, and $\{s_1, \ldots, s_N\}$ a set of observations from that model. A *statistic* $T(s_1, \ldots, s_N)$, i.e., some statistical property of the set, is *sufficient* if it captures all the information relevant to *statistical inference* of the model, i.e.,

$$\Pr(X = x|T(X), \theta) = \Pr(X = x|T(X))$$

▮

An example for sufficient statistics of a normally distributed random variable are its mean and variance. Given a set of observations, the maximum likelihood estimate of the Gaussian process generating these samples is $\Pr(x) \sim N(\mu, \sigma)$ where μ is the mean of the set of observations, and σ is their standard deviation.

In the case of a dynamic process, in each trajectory $(x^{(t_0)}, \ldots, x^{(t_K)})$ we can measure the sufficient statistics are the *residence time* of the process in a state x, denoted as T_x and the *number of transitions* of the process from state x to y, written as $M_{x,y}$. The expectations of these values are the sufficient statistics of any Markov dynamic process. They can both be calculated naively by enumerating over the exponential number of trajectories and averaging over them.

These statistics shine a new light on the alternative representation of the process using the marginals and joint distributions. The connection between the two can be seen in the calculation

$$\begin{aligned}\mathbf{E}\left[T_x\right] &= \sum_{k=0}^{K-1} \mu_x[t_k]\Delta_K \\ \mathbf{E}\left[M_{x,y}\right] &= \sum_{k=0}^{K-1} \mu_{x,y}[t_k, t_{k+1}]\Delta_K \ . \qquad (2.8)\end{aligned}$$

This leads us to see these parameters as the *natural parameterization* of this model, as they are exactly the answers to the inference queries.

For multi component processes, we do not require the statistics of complete assignments $\boldsymbol{x} = (x_1, \ldots, x_D)$. Due to the sparse nature of the process, it is sufficient to have the statistics of each component given each state of its parents. For example, the expected residence times we require are those of each component X_i in a state x_i conditioned on its parents $\mathbf{Pa}_i$ being in state $\boldsymbol{u}_i$, denoted as $\mathbf{E}\left[T_{x_i|\boldsymbol{u}_i}\right]$. Likewise, the conditional expected number of transitions $\mathbf{E}\left[M_{x_i,y_i|\boldsymbol{u}_i}\right]$

Inference in these temporal models is generally a difficult task (Boyen and Koller, 1998).The inference calculations shown above enumerate over all possible states, and because this number is exponential and the summation performed in (2.7) is intractable, and so the usage of approximations is essential.

Approximate Inference

The first approach we will consider is sampling (Koller and Friedman, 2009). Here we must sample trajectories from the distribution induced by the transition probability matrix and the evidence. Due to the dependence between components, we must use the Gibbs sampling procedure, starting with an initial trajectory for each component. In each iteration we select a random component, and sample its new trajectory given the others. As in the Gibbs procedure for joint probabilities, this method generates a chain of samples, and resulting in a set of independent trajectories. The expected sufficient statistics are calculated as the empirical mean of statistics of the generated trajectories.

This MCMC approach is again asymptotically unbiased, but has the same downsides as the Bayesian network Gibbs procedure, namely no error estimate and a long runtime. Similarly, there is a message passing paradigm in discrete Markov processes which is analogous to the algoithms for approximating the marginals in Bayesian networks. This algorithm has similar advantages and disadvantages to its counterpart. We proceed to the next group of algorithms - variational approximations in discrete processes. These will give us a good intuition for our continuous-time approximation.

Variational Approximations in Discrete Markov Processes

Variational approximations in discrete processes resembles those in continuous-time processes. In this section we will show these discrete approximations. The extensions to the continuous case will be shown in the next chapter.

In discrete processes, similarly to those in Bayesian networks, the variational method casts the inference as an optimization problem. We define the optimization procedure over a functional. After defining the constraints we define a Lagrangian and using its partial derivatives we can derive update equations for each $X_i^{(t_0)}$. Similarly to the Bayesian network framework, the constraints we select for the optimization will determine the resulting distribution. Next we will shortly discuss the Mean Field approximation in discrete-time processes.

Mean Field Approximation The following section will briefly describe the discrete case, leaving the details for the continuous-time case. As opposed to the Bayesian network framework, assuming a fully independent distribution throughout the process will lead to a very crude estimate of the distribution. Instead, we will leave the influence of each component on itself in the next time slice, as seen in Fig. 2.4 (b). This can be understood as approximating the complex intertwined process using a set of independent processes. The optimization will find the closest independent set, i.e., the one where each process evolves independently of its peers, yet with a behaviour similar to that its corresponding component in the original process. An important fact is that in general, due to the evidence, our new approximating distribution is now inhomogeneous, i.e., cannot be parameterized using a constant transition probability matrix. This can be understood intuitively by looking at the final transition matrix, which gives a transition probability $p_{x,e_{t_K}}(t_{K-1}) = 1$ from any state $x \neq e_{t_K}$ to e_{t_K}, as $\Pr(X^{(t_K)} = x) = 0$. In the middle of the interval though, as the process is irreducible, we have $p_{x,e_{t_K}}(t_k) < 1$. We can derive this approximation from the discretized Mean Field procedure in continuous-time models. This connection between between the continuous and discrete models will give us an intuition about the approximation for continuous-time models.

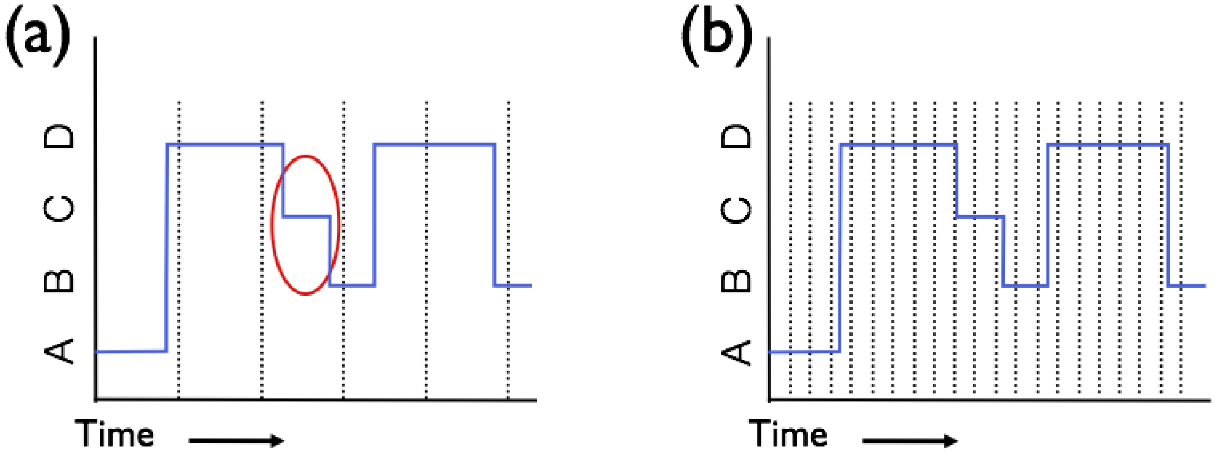

Figure 2.5: Futile attempts to discretize a process using constant time slices with different granularities. **(a)** Resolution is too low. **(b)** Resolution is too high.

2.3 Continuous Time Markov Processes

In the previous section, we depicted the representation of a general stochastic process in discrete time slices. In this section we will present a continuous representation for processes with continuous-time evolution.

Naively, any Markov process can be represented in this fashion by discretizatizing the entire interval into regular time slices of length h. However, in many cases this representation is either inefficient or insufficient. Examples for both these situations can be seen in Fig. 2.5 where in (a) the resolution is too low, resulting in loss of information in the form of an unaccounted for transition. Additionally, even if we start with a sparse network, a high granularity will result in the entanglement of many variables and thus the loss of sparsity. In (b) the resolution is unnecessarily over-accurate, leading to a high computational complexity.

The problem of discretizing continuous processes gives us the required motivation for representing these processes using a "continuous-time language". Due to our process evolving in an infinitesimal time scale, the likelihood of transitioning between the different states is no longer represented using transition probability matrices as in the discrete case. Instead, we describe its continuous dynamics with *transition rates*. The details of this representation will be devulged in the next section.

A trajectory of a continuous-time Markov process $\boldsymbol{X}(t)$ over the interval $[0, T]$ can be parameterized by the *finite* number n of transitions, a set $\{t_1, \dots, t_n\}$ of real-valued transition times and the set $\{x^{(t_0)}, \dots, x^{(t_n)}\}$ of $n + 1$ states. A continuous-time process maps each trajectories and assigns each of them a joint probability

$$\Pr\left(\boldsymbol{x}^{(t_0)}, \ldots, \boldsymbol{x}^{(t_n)}\right) = \Pr\left(\boldsymbol{x}^{(t_0)}\right) \prod_{k=1}^{n} \Pr\left(\boldsymbol{X}^{(t_i)} = x^{(t_i)} | \boldsymbol{X}^{(t_{i-1})} = x^{(t_{i-1})}\right) \quad .$$

In the next section we discuss the representation of single and multiple component processes.

2.3.1 Single Component Process Representation

The dynamics of a continuous-time Markov process are fully captured by a matrix $\mathbb{Q}$—the *rate matrix* with non-negative off-diagonal entries $q_{x,y}$ and diagonal $q_{x,x} = -\sum_{y \neq x} q_{x,y}$. This rate matrix defines the infinitesimal transition probabilities

$$p_{x,y}(h) = \delta_{x,y} + q_{x,y} \cdot h + o(h) \tag{2.9}$$

where $\delta_{x,y}$ is a multivariate Kronecker delta and $o(\cdot)$ means decay to zero faster than its argument. Using the rate matrix $\mathbb{Q}$, we can express the Markov transition function as $p_{x,y}(t) = [\exp(t\mathbb{Q})]_{x,y}$ where $\exp(t\mathbb{Q})$ is a matrix exponential (Chung, 1960; Gardiner, 2004). The exponential is defined for any square matrix $\mathbb{A}$ similarly to the *Talyor series* for the rational exponent

$$\exp\{A\} = \mathbf{I} + \sum_{k=1}^{\infty} \frac{\mathbb{A}^k}{k!} \quad .$$

As the exponential is defined as an infinite series, the actual calculation must be approximate, yet as accurate as possible. The simplest approximation would be to truncate this series at some coefficient, i.e.,

$$\exp\{A\} \approx \mathbf{I} + \sum_{k=1}^{K} \frac{\mathbb{A}^k}{k!} \quad ,$$

which would lead to a significant error in the calculation. The approximation we use in our work is the *Padé* approximation used for general functions Given a function f and two integers $m, n \leq 0$, the *Padé approximant* of order (m, n) is

$$R(x) = \frac{1 + p_1 x + p_2 x^2 + \ldots + p_m x^m}{1 + q_1 x + q_2 x^2 + \ldots + q_n x^n} \quad ,$$

the coefficients $p_0, \ldots, p_m, q_1, \ldots, q_n$ can be determined uniquely. The general formulation of the Padé approximation can be found in Num (2007). The interested reader is referred to Moler and Van Loan (1978) for 18 more dubious ways of calculating a matrix exponential.

Naturally, we will now discuss multi component continuous processes and the challenges they pose.

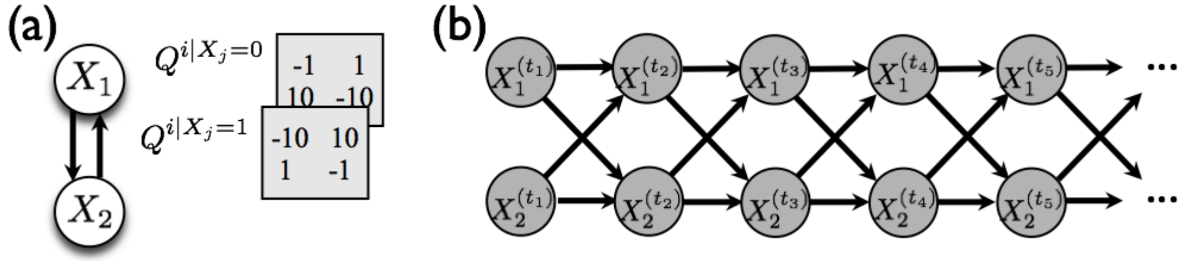
(a)
X_1
X_2
$Q^{i|X_j=0}$
-1 1
10 -10
$Q^{i|X_j=1}$
-10 10
1 -1
(b)
$X_1^{(t_1)}$ $X_1^{(t_2)}$ $X_1^{(t_3)}$ $X_1^{(t_4)}$ $X_1^{(t_5)}$ ...
$X_2^{(t_1)}$ $X_2^{(t_2)}$ $X_2^{(t_3)}$ $X_2^{(t_4)}$ $X_2^{(t_5)}$ ...

Figure 2.6: Two representations of a two binary component dynamic process. **(a)** The associated CTBN and its conditional rate matrices. **(b)** The DBN corresponding to the CTBN in (a). The models are equivalent when $h \to 0$.

2.3.2 Multi-component Representation - Continuous Time Bayesian Networks

A *continuous-time Bayesian network* is defined by assigning each component i a set of components $\mathbf{Pa}_i \subseteq \{1, \dots, D\} \setminus \{i\}$, which are its parents in the network (Nodelman et al., 2002). With each component i we then associate a set of conditional rate matrix $\mathbb{Q}^{i|\mathbf{Pa}_i}_{\cdot|\boldsymbol{u}_i}$ for each state $\boldsymbol{u}_i$ of $\mathbf{Pa}_i$. The off-diagonal entries $q^{i|\mathbf{Pa}_i}_{x_i,y_i|\boldsymbol{u}_i}$ represent the rate at which X_i transitions from state x_i to state y_i given that its parents are in state $\boldsymbol{u}_i$. The diagonal entries are

$$q^{i|\mathbf{Pa}_i}_{x_i,x_i|\boldsymbol{u}_i} = -\sum_{z_i \neq x_i} q^{i|\mathbf{Pa}_i}_{x_i,z_i|\boldsymbol{u}_i}$$

to be the negative of the diagonal elements of the conditional rate matrices (in a rate matrix each row sums up to zero). The dynamics of $\boldsymbol{X}^{(t)}$ are defined by a rate matrix $\mathbb{Q}$ with entries $q_{\boldsymbol{x},\boldsymbol{y}}$, which amalgamates the conditional rate matrices as follows:

$$q_{\boldsymbol{x},\boldsymbol{y}} = \begin{cases} q^{i|\mathbf{Pa}_i}_{x_i,y_i|\boldsymbol{u}_i} & \delta(\boldsymbol{x},\boldsymbol{y}) = \{i\} \\ \sum_i q^{i|\mathbf{Pa}_i}_{x_i,x_i|\boldsymbol{u}_i} & \boldsymbol{x} = \boldsymbol{y} \\ 0 & \text{otherwise,} \end{cases} \tag{2.10}$$

where $\delta(\boldsymbol{x},\boldsymbol{y}) = \{i | x_i \neq y_i\}$. This definition implies that changes are one component at a time.

Example 2.3.1: For clarification, we will study the simple case of a two component binary system with two states, either -1 or 1. Both components have a disposition of being in the same state. This system is called a two component *ising chain* and will be generalized later. For simplification, we assume the system is symmetric, namely that the conditional rate matrices of both components are identical. In Fig. 2.6(a) we can see the structure of the CTBN, and the rate matrices associated with each of its parent's possible assignments.

The relationship between this continuous model's discrete counterpart can be seen in Fig. 2.6(b). The corresponding DBN is discretized with some given h. Each time slice t_k, we have a random variable $X_i(t_k)$ relating to the component X_i. This variable is the parent of the corresponding random variables of its children in the CTBN itself in slice t_{k+1}.

The transition probabilities for each $X_i(t_k)$ are given by the combination of Equations 2.9 and 2.10. From these equations, it can be seen that whenever the components are in different states, they have a higher probability of switching to the state of the other component. Thus, the probability of transitioning to the identical state is much higher than transitioning into the opposite one. ▮

In our framework, we receive evidence of the states of several or all components in our network along some interval $[0, T]$. The two possible types of evidence that may be given are continuous evidence, where we know the state of a subset $\boldsymbol{U} \subseteq \boldsymbol{X}$ continuously over some sub-interval $[t_1, t_2] \subseteq [0, T]$, and point evidence of the state of $\boldsymbol{U}$ at some internal point $t \in [0, T]$. For convenience we restrict our treatment to a time interval $[0, T]$ with end-point evidence $\boldsymbol{X}^{(0)} = \boldsymbol{e}_0$ and $\boldsymbol{X}^{(T)} = \boldsymbol{e}_T$. For components i with continuous evidence $X^{(t)} = x$ we simply assign the constant $\mu_y(t) = \delta_{x,y}$.

The inference queries in continuous-time processes are similar to those in discrete time. These calculated values are needed for the learning of the parameters of the process (Nodelman et al., 2003). For example, the maximum-likelihood estimate for the rate of transition conditioned on the state of $\mathbf{Pa}_i$, denoted as $\hat{q}_{x_i,y_i|\boldsymbol{u}_i}$. Given the expected number of transitions $\mathbf{E}\left[Mx_i, y_i|\boldsymbol{u}_i\right]$ and the the expected residence time $\mathbf{E}\left[T_{x_i|\boldsymbol{u}_i}\right]$ corresponding to the query, this estimate can be written as

$$\hat{q}_{x_i,y_i|\boldsymbol{u}_i} = \frac{\mathbf{E}\left[M_{x_i,y_i|\boldsymbol{u}_i}\right]}{\mathbf{E}\left[T_{x_i|\boldsymbol{u}_i}\right]} \quad .$$

We will discuss the calculations of these queries in the next section.

2.3.3 Inference in Continuous Time Markov Processes

Given a continuous-time Bayesian network, we would like to evaluate the likelihood of evidence, to compute the probability of various events given the evidence (e.g., that the state of the system at time t is $\boldsymbol{x}$), and to compute conditional expectations (e.g., the expected amount of time X_i was in state x_i conditioned on some state of $\mathbf{Pa}_i$). Every continuous-time process is fully described by the transition rate matrix, and its *sufficient statistics*, which as illustrated above are essential for learning the parameters of the process. These queries are much like those of discrete processes, namely the expected residence times $\mathbf{E}\left[T_{x_i|\boldsymbol{u}_i}\right]$ and the expected number of transitions $\mathbf{E}\left[M_{x_i,y_i|\boldsymbol{u}_i}\right]$. Additionally, we need the marginal probabilities $\mu^i_{x_i|\boldsymbol{u}_i}(t) = \Pr(X_i(t) = x_i|\boldsymbol{U}_i(t) = \boldsymbol{u}_i)$. These statistics are dependent on the probability is induced from the rate matrix $\mathbb{Q}$ and the evidence.

Exact inference

Incidentally, calculation of these values is usually an intractable task. For instance, the calculation of the marginals is given by the exponent

$$\begin{aligned}\mu_{\boldsymbol{x}}(t) &= \left[\mu_{\boldsymbol{e}_0}(0) \cdot \exp\left\{\mathbb{Q} \cdot t\right\}\right]_{\boldsymbol{e}_0,\boldsymbol{x}} \\ \mu^{i}_{x_i|\boldsymbol{u}_i}(t) &= \sum_{\boldsymbol{x}\setminus\{i,\boldsymbol{u}_i\}} \mu_{\boldsymbol{x}}(t) \ .\end{aligned}$$

The expected residence times and transitions can be calculated as

$$\begin{aligned}\mathbf{E}\left[T_{x_i|\boldsymbol{u}_i}\right] &= \int_0^T \mu_{x_i|\boldsymbol{u}_i}(t)dt \\ \mathbf{E}\left[M_{x_i,y_i|\boldsymbol{u}_i}\right] &= \int_0^T \mu_{x_i|\boldsymbol{u}_i}(t)q_{x_i,y_i|\boldsymbol{u}_i}dt \ .\end{aligned}$$

Direct computations of these quantities involve matrix exponentials of the rate matrix $\mathbb{Q}$, whose size is exponential in the number of components, and using it for integration over the the time interval. These make this approach infeasible beyond a modest number of components. We therefore have to resort to approximations.

Gibbs sampling

Similarly to the discrete Markov processes, the statistics in continuous-time processes can be calculated using sampling techniques which sample trajectories in systems with dependencies (El-Hay et al., 2008). The main difference is that here the transitions of the randomly chosen component are distributed exponentially, depending on the state of its parents. Sampling enough trajectories can allow us to calculate these values by simply using the empirical mean as our estimate. For example, the residence time can be calculated by

$$\mathbf{E}_{\mathbf{P}}\left[\boldsymbol{T}_{x_i|\boldsymbol{u}_i}\right] = \frac{1}{N}\sum_{k=1}^{N}\boldsymbol{T}^{k}_{x_i|\boldsymbol{u}_i}$$

where $\boldsymbol{T}^{k}_{x_i|\boldsymbol{u}_i}$ is the residence time of the i^{th} component in state x_i given its parents are in state $\boldsymbol{u}_i$, as sampled in the k^{th} trajectory.

Expectation propagation

The *Expectation propagation* algorithm, introduced in Nodelman et al. (2005b) approximates the continuous process using a piece-wise homogeneous approximation. It is a message passing algortihm, and so models the dependence between the different components, but only requires a local consistency, much like in the discrete case. The regions in the algorithm do not contain distributions over the region variables at individual time points, but distributions over trajectories of the variables throughout the domain. This it can adapt the time granularity of reasoning for different variables and in different condtions.

The algorithm's disadvantages are similar to the discrete message passing algorithms, namely that it has no convergence guaranties and, as opposed to the Mean Field approximation we will show next, does not give a lower bound on the log-likelihood of the evidence. It also requires a tuning of parameters, which we would like to avoid and have the algorithm perform the tuning adaptively.

Chapter 3

Variational Approximations in Continuous-Time Bayesian Networks

3.1 Variational Principle

We start by defining a variational approximation principle in terms of a general continuous-time Markov process (that is, without assuming any network structure). Here we aim to define a lower bound on $\ln P_{\mathbb{Q}}(\boldsymbol{e}_T|\boldsymbol{e}_0)$ as well as to approximate the posterior probability $P_{\mathbb{Q}}(\cdot \mid \boldsymbol{e}_0, \boldsymbol{e}_T)$.

Variational approximations cast inference as an optimization problem of a functional which approximates the log probability of the evidence by introducing an auxiliary set of *variational parameters*. The posterior distribution of a Markov process can be represented in several ways. Before we define this representation we will first examine its structure.

Inhomogeneous Rates Representation As discussed above, the prior distribution of the process can be characterized by a time-independent rate matrix $\mathbb{Q}$. It is easy to show that if the prior is a Markov process, then the posterior $\Pr(\cdot|\boldsymbol{e}_0, \boldsymbol{e}_T)$ is also a Markov process, albeit not necessarily a homogeneous one. Such a process can be represented by a time-dependent rate matrix $\mathbb{R}(t)$ that describes the instantaneous transition rates. This approximation, although intuitive, proves problematic in the framework of deterministic evidence, as can be seen in the following example.

Example 3.1.1: To illustrate how the rates diverge as we get closer to the end of the interval we take a simple system with one binary component, and evidence $e_0 = A$, $e_T = B$. Intuitively, any trajectory that is still in state A when t approaches T will need to be pushed harder and harder to state B and at the limit, this rate will have to be infinite to assure that $\mu_A(T) = 0$. More formally, we can write the rate of transition

$$\mathbb{R}_{A,B}(t) = \lim_{h \to 0} \frac{\Pr(X(t) = B|X(t-h) = A, e_T)}{h} \quad .$$

Taking the rates to $t = T$ we get

$$\mathbb{R}_{A,B}(T) = \lim_{h \to 0} \frac{\Pr(X(T) = B|X(t-h) = A, e_T)}{h} \quad .$$

Due to the deterministic evidence, the numerator tends to the constant 1 as we approach $t = T$. On the other hand, the denominator obviously goes to 0, giving us divergent rates. While this problem occurs in our framework where we have "hard" evidence (the probabilities of the different states at the observed points are either 0 or 1, this does not occur when the evidence is noisy, and this representation can be useful (Opper and Sanguinetti, 2007). In our framework, we therefore have to find an alternative parameterization for this inhomogeneous process.

While this parameterization may seem problematic, we noticed that whenever the rates $\mathbb{R}_{A,B}(t)$ diverge as $t \to T$, the marginals $\mu_A(t)$ must go to zero. Another non trivial fact is that the multiplication of the marginal and the rate $\mu_A(t) \cdot \mathbb{R}_{A,B}(t)$ never diverges, and so this parameter is numerically stable. When $A \neq B$ this can be shown by decomposing this parameter as a *joint probability density*

$$\mu_A(t) \cdot \mathbb{R}_{A,B}(t) = \lim_{h \to 0} \frac{\Pr(X^{(t)} = A, X^{(t+h)} = B | e_0, e_T)}{h}$$

and so writing this density at $t = T$ as the posterior probability of a Markov process

$$\begin{aligned} & \frac{\Pr(X^{(T-h)} = A, X^{(T)} = B | e_0, e_T)}{h} \\ = \quad & \frac{\Pr(X^{(T-h)} = A | e_0) \cdot \Pr(X^{(T)} = B | X^{(T-h)} = A) \Pr(e_T | X^{(T)} = B)}{\Pr(e_T | e_0) \cdot h} \\ = \quad & \frac{\Pr(X^{(T-h)} = A | \boldsymbol{e}_0) \cdot q_{A,B} \cdot h}{\Pr(e_T | e_0) \cdot h} \end{aligned}$$

which has no infinite terms. In the case where $A = B$, we have that $\mu_A(T) \cdot R_{A,A}(t) = -\sum_{B \neq A} \mu_A(T) \cdot R_{A,B}(t)$, which is a simple sum, and so does not diverge. ▮

This motivation leads us to defining a new set of numerically stable variational parameters.

Marginal Density Representation As we have seen, continuous-time models require a slightly modified parametrization for variational inference. Here we define the optimization problem over a set of *mean parameters* (Wainwright and Jordan, 2008), representing possible values of expected sufficient statistics. Here, rather than representing the posterior distribution by a time-dependent rate matrix, we now consider a representation that is more natural for variational approximations. Let $\Pr$ be the distribution of a Markov process. We define a family of functions:

$$\begin{aligned} \mu_{\boldsymbol{x}}(t) &= \Pr(\boldsymbol{X}^{(t)} = \boldsymbol{x}) \\ \gamma_{\boldsymbol{x},\boldsymbol{y}}(t) &= \lim_{h \downarrow 0} \frac{\Pr(\boldsymbol{X}^{(t)} = \boldsymbol{x}, \boldsymbol{X}^{(t+h)} = \boldsymbol{y})}{h}, \quad \boldsymbol{x} \neq \boldsymbol{y} \\ \gamma_{\boldsymbol{x},\boldsymbol{x}}(t) &= -\sum_{\boldsymbol{y} \neq \boldsymbol{x}} \gamma_{\boldsymbol{x},\boldsymbol{y}}(t). \end{aligned} \tag{3.1}$$

The function $\mu_{\boldsymbol{x}}(t)$ is the probability that $\boldsymbol{X}^{(t)} = \boldsymbol{x}$. The function $\gamma_{\boldsymbol{x},\boldsymbol{y}}(t)$ is the probability density that $\boldsymbol{X}$ transitions from state $\boldsymbol{x}$ to $\boldsymbol{y}$ at time t. Note that this parameter is not a transition rate, but rather a product of a point-wise probability with the point-wise transition rate of the approximating probability, i.e., the time-dependent rate matrix is defined $\mathbb{R}_{x,y}(t) = \gamma_{x,y}(t)/\mu_x(t)$. Hence, unlike the (inhomogeneous) rate matrix at time t, $\gamma_{x,y}(t)$ takes into account the probability of being in state $\boldsymbol{x}$ and not only the rate of transitions.

Proposition 3.1.2:

$$\Pr(\boldsymbol{X}^{(t)} = \boldsymbol{x}, \boldsymbol{X}^{(t+h)} = \boldsymbol{y}) = \mu_{\boldsymbol{x}}(t)\delta_{\boldsymbol{x},\boldsymbol{y}} + \gamma_{\boldsymbol{x},\boldsymbol{y}}(t)h + o(h), \tag{3.2}$$

Proof: In the case that $\boldsymbol{y} \neq \boldsymbol{x}$, this is the exact definition of $\gamma_{\boldsymbol{x},\boldsymbol{y}}(t)$. In the case where $\boldsymbol{y} = \boldsymbol{x}$ we can write

$$\begin{aligned}\Pr(X^{(t)} = \boldsymbol{x}, X^{(t+h)} = \boldsymbol{x}) &= \mu_{\boldsymbol{x}}(t) - \sum_{\boldsymbol{y}\neq\boldsymbol{x}} \Pr(X^{(t)} = \boldsymbol{x}, X^{(t+h)} = \boldsymbol{y}) + o(h) \\ &= \mu_{\boldsymbol{x}}(t) + \gamma_{\boldsymbol{x},\boldsymbol{x}}(t)h + o(h) \ .\end{aligned}$$

▮

We aim to use the family of functions μ and γ as a representation of a Markov process. To do so, we need to characterize the set of constraints that these functions should satisfy.

Definition 3.1.3: A family $\eta = \{\mu_{\boldsymbol{x}}(t), \gamma_{\boldsymbol{x},\boldsymbol{y}}(t) : 0 \leq t \leq T\}$ of continuous functions is a *Markov-consistent density set* if the following constraints are fulfilled:

$$\begin{aligned}\mu_{\boldsymbol{x}}(t) &\geq 0, \quad \sum_{\boldsymbol{x}} \mu_{\boldsymbol{x}}(0) = 1, \\ \gamma_{\boldsymbol{x},\boldsymbol{y}}(t) &\geq 0 \qquad \forall \boldsymbol{y} \neq \boldsymbol{x}, \\ \gamma_{\boldsymbol{x},\boldsymbol{x}}(t) &= -\sum_{\boldsymbol{y}\neq\boldsymbol{x}} \gamma_{\boldsymbol{x},\boldsymbol{y}}(t), \\ \frac{d}{dt}\mu_{\boldsymbol{x}}(t) &= \sum_{\boldsymbol{y}} \gamma_{\boldsymbol{y},\boldsymbol{x}}(t).\end{aligned}$$

Let $\mathcal{M}$ be the set of all Markov-consistent densities. ▮

As an analogy to the discrete case, we will compare this parameterization to the "natural" parameterization of variational inference in dynamic Bayesian networks of Equation (2.8). First, in both cases we constrain the marginals $\mu_{\boldsymbol{x}}(t)$ to be valid distributions. Additionally, in the discrete case we constrain the joint distributions over two time slices $\mu_{\boldsymbol{z},\boldsymbol{x}}[t_k, t_{k+1}]$ to be non-negative and agree on their intersecting time slice. The marginals of that time slice $\mu_{\boldsymbol{x}}[k]$ are equal to the marginalization over either of the two joint probability distributions, i.e.,

$$\mu_{\boldsymbol{x}}[t_k] = \sum_{\boldsymbol{z}} \mu_{\boldsymbol{z},\boldsymbol{x}}[t_{k-1}, t_k] = \sum_{\boldsymbol{z}} \mu_{\boldsymbol{x},\boldsymbol{z}}[t_k, t_{k+1}]$$

In the continuous case with infinitesimal discretization, we get the similar equation

$$\mu_{\boldsymbol{x}}(t) = \mu_{\boldsymbol{x}}(t-h) + \sum_{\boldsymbol{y}} \gamma_{\boldsymbol{y},\boldsymbol{x}}(t-h) \cdot h$$

by the same marginalization, but with the small difference justified in (3.2). Rearranging the terms and going to the limit $h \to 0$ gives us the time-derivative of $\mu_{\boldsymbol{x}}(t)$. This derivative, along with the initial values $\mu_{\boldsymbol{x}}(0) = \delta_{\boldsymbol{x},\boldsymbol{e}_0}$ gives us the *forward master equation* of the marginals.

Using standard arguments we can show that there exists a correspondence between (generally inhomogeneous) Markov processes and density sets η. Specifically:

Lemma 3.1.4: *Let $\eta = \{\mu_{\boldsymbol{x}}(t), \gamma_{\boldsymbol{x},\boldsymbol{y}}(t)\}$. If $\eta \in \mathcal{M}$, then there exists a continuous-time Markov process P_η for which $\mu_{\boldsymbol{x}}$ and $\gamma_{\boldsymbol{x},\boldsymbol{y}}$ satisfy (3.1).*

Proof: Given η, we define the *inhomogeneous rate matrix* $\mathbb{R}_{x,y}(t) = \frac{\gamma_{x,y}(t)}{\mu_x(t)}$ wherever $\mu(t) > 0$, and for the singular points where $\mu_x(t) = 0$ $R_{x,y}(t) = 0$. From its definition $R(t)$ is a valid rate matrix - its non-diagonals are non-negative as they are the quotient of two non-negative functions, and the requirements of definition (3.1)

$$R_{x,x}(t) = \frac{\gamma_{x,x}(t)}{\mu_x(t)} = -\frac{\gamma_{x,y}(t)}{\mu_x(t)} = -\sum_y R_{x,y}(t) \quad .$$

Thus R's diagonals are negative and its rows sum to 0. From Chung (1960) we can use these rates with the initial value $\mu_x(0)$ to construct the Markov process P_η and its probability $\mathbf{P}(t)$ from the forward master equation

$$\frac{d}{dt} P_\eta = \mathbb{R}(t) P_\eta$$

and

$$P_\eta(0) = \mu(0) \quad .$$

Finally, to satisfy (3.1) we have to prove that $\mu_x(t)$ are the true marginals and that $\gamma_{x,y}(t)$ equal the joint probability densities. First, because of the matching initial value at $t = 0$ and the equality of the time-derivatives of the two functions, thus

$$\mu_x(t) = \mathbf{P}_x(t) = \Pr(X^{(t)} = x) \quad .$$

Next, the equivalence of the joint probability densities can be proved:

$$\begin{aligned} \lim_{h \to 0} \frac{\Pr(X^{(t)} = x, X^{(t+h)} = y)}{h} &= \lim_{h \to 0} \frac{\mu_x(t) \Pr(X^{(t+h)} = y | \Pr(X^{(t)} = x)}{h} \\ &= \lim_{h \to 0} \frac{\mu_x(t) \mathbb{R}_{x,y}(t) h}{h} \\ &= \mu_x(t) \mathbb{R}_{x,y}(t) \end{aligned}$$

which is exactly $\gamma_{x,y}(t)$. ∎

The processes we are interested in, however, have additional structure, as they correspond to the posterior distribution of a time-homogeneous process with end-point evidence. This additional structure implies that we should only consider a subset of $\mathcal{M}$:

Lemma 3.1.5: *Let $\mathbb{Q}$ be a rate matrix, and $\boldsymbol{e}_0, \boldsymbol{e}_T$ be states of $\boldsymbol{X}$. Then the representation η corresponding to the posterior distribution $P_{\mathbb{Q}}(\cdot|\boldsymbol{e}_0, \boldsymbol{e}_T)$ is in the set $\mathcal{M}_{\boldsymbol{e}} \subset \mathcal{M}$ that contains Markov-consistent density sets $\{\mu_{\boldsymbol{x}}(t), \gamma_{\boldsymbol{x},\boldsymbol{y}}(t)\}$, satisfying $\mu_{\boldsymbol{x}}(0) = \delta_{\boldsymbol{x},\boldsymbol{e}_0}$, $\mu_{\boldsymbol{x}}(T) = \delta_{\boldsymbol{x},\boldsymbol{e}_T}$.*

Thus, from now on we can restrict our attention to density sets from $\mathcal{M}_{\boldsymbol{e}}$. We can now state the variational principle for continuous processes, which closely tracks similar principles for discrete processes. Here we define a *free energy functional*,

$$\mathcal{F}(\eta; \mathbb{Q}) = \mathcal{E}(\eta; \mathbb{Q}) + \mathcal{H}(\eta),$$

which, as we will see, measures the quality of η as an approximation of $P_{\mathbb{Q}}(\cdot|\boldsymbol{e})$. (For succinctness, we will assume that the evidence $\boldsymbol{e}$ is clear from the context.) The two terms in the continuous functional correspond to an entropy,

$$\mathcal{H}(\eta) = \int_0^T \sum_{\boldsymbol{x}} \sum_{\boldsymbol{y} \neq \boldsymbol{x}} \gamma_{\boldsymbol{x},\boldsymbol{y}}(t)[1 + \ln \mu_{\boldsymbol{x}}(t) - \ln \gamma_{\boldsymbol{x},\boldsymbol{y}}(t)]dt,$$

and an energy,

$$\mathcal{E}(\eta; \mathbb{Q}) = \int_0^T \sum_{\boldsymbol{x}} \left[\mu_{\boldsymbol{x}}(t) q_{\boldsymbol{x},\boldsymbol{x}} + \sum_{\boldsymbol{y} \neq \boldsymbol{x}} \gamma_{\boldsymbol{x},\boldsymbol{y}}(t) \ln q_{\boldsymbol{x},\boldsymbol{y}} \right] dt.$$

To perform optimization of this functional, we will next prove its relation to the KL divergence and the likelihood of the evidence, and thus casts the variational inference into an optimization problem.

Theorem 3.1.6: Let $\mathbb{Q}$ be a rate matrix, $\boldsymbol{e} = (\boldsymbol{e}_0, \boldsymbol{e}_T)$ be states of $\boldsymbol{X}$, and $\eta \in \mathcal{M}_{\boldsymbol{e}}$. Then

$$\mathcal{F}(\eta; \mathbb{Q}) = \ln P_{\mathbb{Q}}(\boldsymbol{e}_T|\boldsymbol{e}_0) - \boldsymbol{D}(P_\eta \| P_{\mathbb{Q}}(\cdot|\boldsymbol{e}))$$

where P_η is the distribution corresponding to η and $\boldsymbol{D}(P_\eta \| P_{\mathbb{Q}}(\cdot|\boldsymbol{e}))$ is the KL divergence between the two processes.

Proof: The basic idea is to consider discrete approximations of the functional. Let K be an integer. We define the K-sieve $\boldsymbol{X}_K$ to be the set of random variables $\boldsymbol{X}^{(t_0)}, \boldsymbol{X}^{(t_1)}, \ldots, \boldsymbol{X}^{(t_K)}$ where $t_k = \frac{kT}{K}$. We can use the variational principle (Jordan et al., 1998) on the marginal distributions $P_{\mathbb{Q}}(\boldsymbol{X}_K|\boldsymbol{e})$ and $P_\eta(\boldsymbol{X}_K)$. More precisely, define

$$\mathcal{F}_K(\eta; \mathbb{Q}) = \mathbf{E}_{P_\eta}\left[\ln \frac{P_{\mathbb{Q}}(\boldsymbol{X}_K, \boldsymbol{e}_T \mid \boldsymbol{e}_0)}{P_\eta(\boldsymbol{X}_K)} \right],$$

which can, by using simple arithmetic manipulations be recast as

$$\mathcal{F}_K(\eta; \mathbb{Q}) = \ln P_{\mathbb{Q}}(\boldsymbol{e}_T|\boldsymbol{e}_0) - \boldsymbol{D}(P_\eta(\boldsymbol{X}_K) \| P_{\mathbb{Q}}(\boldsymbol{X}_K|\boldsymbol{e})).$$

We get the desired result by letting $K \to \infty$. By definition $\lim_{K\to\infty} \boldsymbol{D}(P_\eta(\boldsymbol{X}_K) \| P_{\mathbb{Q}}(\boldsymbol{X}_K|\boldsymbol{e}))$ is $\boldsymbol{D}(P_\eta(\cdot) \| P_{\mathbb{Q}}(\cdot|\boldsymbol{e}))$ and using the following Lemma 3.1.7 gives the result. ∎

Lemma 3.1.7: $\mathcal{F}(\eta;\mathbb{Q}) = \lim_{K\to\infty}\mathcal{F}_K(\eta;\mathbb{Q})$.

Proof: Since both $P_{\mathbb{Q}}$ and P_η are Markov processes, we can write

$$\begin{aligned}\mathcal{F}_K(\eta;\mathbb{Q}) = & \sum_{k=0}^{K-1} \mathbf{E}_{P_\eta}\left[\ln P_{\mathbb{Q}}(\boldsymbol{X}^{(t_{k+1})}|\boldsymbol{X}^{(t_k)})\right] \\ & + \sum_{k=0}^{K-1} \mathbf{E}_{P_\eta}\left[\ln P_\eta(\boldsymbol{X}^{(t_k)},\boldsymbol{X}^{(t_{k+1})})\right] \\ & - \sum_{k=1}^{K-1} \mathbf{E}_{P_\eta}\left[\ln P_\eta(\boldsymbol{X}^{(t_k)})\right]\end{aligned}$$

We now express these terms as functions of $\mu_{\boldsymbol{x}}(t)$, $\gamma_{\boldsymbol{x},\boldsymbol{y}}(t)$ and $q_{\boldsymbol{x},\boldsymbol{y}}$. By definition, $P_\eta(\boldsymbol{X}^{(t_k)} = \boldsymbol{x}) = \mu_{\boldsymbol{x}}(t_k)$. Each of the expectations either depend on this term, or on the joint distribution $P_\eta(\boldsymbol{X}^{(t_{k-1})},\boldsymbol{X}^{(t_k)})$. Using the continuity of $\gamma_{\boldsymbol{x},\boldsymbol{y}}(t)$ we can rewrite the probability of such joint events as

$$P_\eta(\boldsymbol{X}^{(t_k)} = \boldsymbol{x}, \boldsymbol{X}^{(t_{k+1})} = \boldsymbol{y}) = \delta_{\boldsymbol{x},\boldsymbol{y}}\mu_{\boldsymbol{x}}(t) + \Delta_K \cdot \gamma_{\boldsymbol{x},\boldsymbol{y}}(t) + o(\Delta_K)$$

where $\Delta_K = T/K$. Similarly, we can also write

$$P_{\mathbb{Q}}(\boldsymbol{X}^{(t_{k+1})} = \boldsymbol{y}|\boldsymbol{X}^{(t_k)} = \boldsymbol{x}) = \delta_{\boldsymbol{x},\boldsymbol{y}} + \Delta_K \cdot q_{\boldsymbol{x},\boldsymbol{y}} + o(\Delta_K) \ .$$

Plugging in these equalities we get

$$\begin{aligned}\mathcal{F}_K(\eta;\mathbb{Q}) \ = & \ \sum_{k=0}^{K-1}\sum_{\boldsymbol{x}}(\mu_{\boldsymbol{x}}(t_k) + \gamma_{\boldsymbol{x},\boldsymbol{x}}(t_k)\Delta_K)\ln\frac{(1+q_{\boldsymbol{x},\boldsymbol{x}}\Delta_K)}{\mu_{\boldsymbol{x}}(t_k) + \gamma_{\boldsymbol{x},\boldsymbol{x}}(t_k)\Delta_K} \\ + & \ \Delta_K \sum_{\boldsymbol{y}\neq\boldsymbol{x}} \gamma_{\boldsymbol{x},\boldsymbol{y}}(t_k)\ln\frac{q_{\boldsymbol{x},\boldsymbol{y}}}{\gamma_{\boldsymbol{x},\boldsymbol{y}}(t_k)} \\ + & \ \sum_{\boldsymbol{x}}\mu_{\boldsymbol{x}}(t_1)\ln\frac{P_{\mathbb{Q}}(\boldsymbol{X}^{(t_1)} = \boldsymbol{x}|\boldsymbol{e}_0)}{\mu_{\boldsymbol{x}}(t_1)} + \sum_{\boldsymbol{x}}\mu_{\boldsymbol{x}}(t_{K-1})\ln P_{\mathbb{Q}}(\boldsymbol{e}_{t_K}|\boldsymbol{X}^{(t_{K-1})} = \boldsymbol{x}) \ .\end{aligned}$$

To further simplify the functional, we define the boundary terms

$$\mathcal{A}_K = \sum_{\boldsymbol{x}}\mu_{\boldsymbol{x}}(t_1)\ln\frac{P_{\mathbb{Q}}(\boldsymbol{X}^{(t_1)} = \boldsymbol{x}|\boldsymbol{e}_0)}{\mu_{\boldsymbol{x}}(t_1)} + \sum_{\boldsymbol{x}}\mu_{\boldsymbol{x}}(t_{K-1})\ln P_{\mathbb{Q}}(\boldsymbol{e}_{t_K}|\boldsymbol{X}^{(t_{K-1})} = \boldsymbol{x}) \tag{3.3}$$

and rewrite $\mathcal{F}_K(\eta;\mathbb{Q})$

$$\begin{aligned}\mathcal{F}_K(\eta;\mathbb{Q}) \ = & \ \sum_{k=0}^{K-1}\sum_{\boldsymbol{x}}(\mu_{\boldsymbol{x}}(t_k) + \gamma_{\boldsymbol{x},\boldsymbol{x}}(t_k)\Delta_K)\ln\frac{(1+q_{\boldsymbol{x},\boldsymbol{x}}\Delta_K)}{\mu_{\boldsymbol{x}}(t_k) + \gamma_{\boldsymbol{x},\boldsymbol{x}}(t_k)\Delta_K} \\ + & \ \Delta_K \sum_{\boldsymbol{y}\neq\boldsymbol{x}} \gamma_{\boldsymbol{x},\boldsymbol{y}}(t_k)\ln\frac{q_{\boldsymbol{x},\boldsymbol{y}}}{\gamma_{\boldsymbol{x},\boldsymbol{y}}(t_k)} + \mathcal{A}_K \ .\end{aligned}$$

Using properties of logarithms we have that

$$\ln\left(1+\Delta_K\cdot z+o(\Delta_K)\right) \;=\; \Delta_K\cdot z+o(\Delta_K).$$

which simplifies the first sum:

$$(\mu_x(t_k)+\gamma_{x,x}(t_k)\Delta_K)\ln(1+q_{x,x}\Delta_K)=\mu_x(t_k)q_{x,x}\Delta_K+o(\Delta_K)$$

and

$$\begin{aligned}
&(\mu_x(t_k)+\gamma_{x,x}(t_k)\Delta_K)\ln(\mu_x(t_k)+\gamma_{x,x}(t_k)\Delta_K)\\
&=\;(\mu_x(t_k)+\gamma_{x,x}(t_k)\Delta_K)\ln\left(1+\frac{\gamma_{x,x}(t_k)\Delta_K}{\mu_x(t_k)}\right)+\mu_x(t_k)\ln\mu_x(t_k)+\gamma_{x,x}(t_k)\Delta_K\ln\mu_x(t_k)\\
&=\;\gamma_{x,x}(t_k)\Delta_K+\mu_x(t_k)\ln\mu_x(t_k)+\gamma_{x,x}(t_k)\Delta_K\ln\mu_x(t_k)+o(\Delta_K)\;\;.
\end{aligned}$$

Our functional can now be written as

$$\begin{aligned}
\mathcal{F}_K(\eta;\mathbb{Q}) \;&=\; \sum_{k=0}^{K-1}\sum_{\boldsymbol{x}}\sum_{\boldsymbol{y}\neq\boldsymbol{x}}\gamma_{\boldsymbol{x},\boldsymbol{y}}(t_k)\left[1+\ln\mu_{\boldsymbol{x},\boldsymbol{y}}(t)-\ln\gamma_{\boldsymbol{x},\boldsymbol{y}}(t_k)\right]\Delta_K\\
&+\;\sum_{k=0}^{K-1}\sum_{\boldsymbol{x}}\left[\mu_{\boldsymbol{x}}(t_k)q_{\boldsymbol{x},\boldsymbol{x}}+\sum_{\boldsymbol{y}\neq x}\gamma_{\boldsymbol{x},\boldsymbol{y}}(t_k)\ln q_{\boldsymbol{x}\boldsymbol{y}}(t_k)\right]\Delta_K+o(\Delta_K)+\mathcal{A}_K\;\;.
\end{aligned}$$

Now we can decompose $\mathcal{F}_K(\eta;\mathbb{Q})$ as the sum of two terms

$$\mathcal{F}_K(\eta;\mathbb{Q})=\mathcal{E}_K(\eta;\mathbb{Q})+\mathcal{H}_K(\eta)+\mathcal{A}_K,$$

where

$$\mathcal{E}_K(\eta;\mathbb{Q})=\sum_{k=0}^{K-1}\Delta_K e_K(t_k),\quad \mathcal{H}_K(\eta)=\sum_{k=0}^{K-1}\Delta_K h_K(t_k),$$

and

$$\begin{aligned}
e_K(t)&=\sum_{\boldsymbol{x}}\sum_{\boldsymbol{y}\neq\boldsymbol{x}}\gamma_{\boldsymbol{x},\boldsymbol{y}}(t)[1+\ln\mu_{\boldsymbol{x}}(t)-\ln\gamma_{\boldsymbol{x},\boldsymbol{y}}(t)]+o(\Delta_K)\\
h_k(t)&=\sum_{\boldsymbol{x}}\left[\mu_{\boldsymbol{x}}(t)q_{\boldsymbol{x}\boldsymbol{x}}+\sum_{\boldsymbol{y}\neq\boldsymbol{x}}\gamma_{\boldsymbol{x},\boldsymbol{y}}(t)\ln q_{\boldsymbol{x},\boldsymbol{y}}\right]+o(\Delta_K)
\end{aligned}$$

Letting $K\to\infty$ we have that $\sum_k\Delta_k[f(t_k)+o(\Delta_K)]\to\int_0^T f(t)dt$, hence $E_K(\eta;\mathbb{Q})$ and $\mathcal{H}_K(\eta)$ converge to $E(\eta;\mathbb{Q})$ and $\mathcal{H}(\eta)$, respectively. According to Lemma 3.1.8, the terms $\mathcal{A}_K$ vanish, and the proof is concluded. ▮

Lemma 3.1.8: *The terms $\mathcal{A}_K$, as defined in (3.3) satisfy*

$$\lim_{\Delta_K\to 0}\mathcal{A}_K=0\;\;.$$

Proof: The first term, involving e_0, goes to zero due to the fact that $\mu_{\boldsymbol{x}}(t_0)$ is a linear approximation of $p_{\boldsymbol{e}_0,\boldsymbol{x}}(\Delta_K)$. In the second, if $\boldsymbol{x} \neq \boldsymbol{e}_{t_K}$ then

$$\mu_{\boldsymbol{x}}(t_{K-1}) \ln P_{\mathbb{Q}}(\boldsymbol{e}_{t_K}|\boldsymbol{X}^{(t_{K-1})} = \boldsymbol{x}) = \frac{P_\eta(\boldsymbol{X}^{(t_{K-1})}|\boldsymbol{e}_0)}{P_\eta(\boldsymbol{e}_{t_K}|\boldsymbol{e}_0)} P_\eta(\boldsymbol{e}_{t_K}|\boldsymbol{X}^{(t_{K-1})}) \ln P_{\mathbb{Q}}(\boldsymbol{e}_{t_K}|\boldsymbol{X}^{(t_{K-1})})$$

and as $\lim_{\Delta_K \to 0} P_\eta(\boldsymbol{e}_{t_K}|\boldsymbol{X}^{(t_{K-1})}) = \lim_{\Delta_K \to 0} P_{\mathbb{Q}}(\boldsymbol{e}_{t_K}|\boldsymbol{X}^{(t_{K-1})}) = 0$, this whole term goes to 0. Conversly, when $\boldsymbol{x} = \boldsymbol{e}_{t_K}$ then $\mu_{\boldsymbol{x}}(t_{K-1})$ goes to 1 as $\Delta_K \to 0$, as does $P_{\mathbb{Q}}(\boldsymbol{e}_{t_K}|\boldsymbol{X}^{(t_{K-1})} = \boldsymbol{x})$, making the logarithm of this term go to 0. ▮

Thm. 3.1.6 allows us to view variational inference as an optimization procedure. In particular we see that the energy functional $\mathcal{F}(\eta; \mathbb{Q})$ is a lower bound of the log-likelihood of the evidence, and the closer the approximation to the target posterior, the tighter the bound. Additionally, once we have found the best approximation, we can calculate the sufficient statistics we need from the result of the optimization.

Taking the general perspective of Wainwright and Jordan (2008), the representation of the distribution uses the natural sufficient statistics. In a Markov process, these are $T_{\boldsymbol{x}}$, the time spent in state $\boldsymbol{x}$, and $M_{\boldsymbol{x},\boldsymbol{y}}$, the number of transitions from state $\boldsymbol{x}$ to $\boldsymbol{y}$. In a discrete-time model, we can capture the statistics for every random variable as in (2.8). In a continuous-time model, however, we need to consider the time derivative of the statistics. Indeed, it is not hard to show that

$$\frac{d}{dt}\mathbf{E}\left[T_{\boldsymbol{x}}(t)\right] = \mu_{\boldsymbol{x}}(t) \quad \text{and} \quad \frac{d}{dt}\mathbf{E}\left[M_{\boldsymbol{x},\boldsymbol{y}}(t)\right] \ = \gamma_{\boldsymbol{x},\boldsymbol{y}}(t).$$

Thus, our marginal density sets η provide what we consider a natural formulation for variational approaches to continuous-time Markov processes. and extracting these values from the resulting approximation requires a simple integration.

Before moving on to the formal definition of the optimization procedure, we will give an intuition of the optimization using the simplistic model of a single component process. This simple case will be generalized later on.

Example 3.1.9: Consider the case of a single component, for which our procedure should be exact, as no simplifying assumptions are made on the density set. Calculation of the sufficient statistics and marginals should be an easy task using *dynamic programming*. A simple strategy for doing this is to first calculate the backward probability $\beta_x(t) = \Pr(e_T|X(t) = x)$, given the end state of the evidence e_T. This probability is derived by using the discretized backward propagation rule

$$\beta_x(t) = \sum_y \Pr(X^{(t+h)} = y|X^{(t)} = x)\beta_y(t+h)$$

which can be made continuous and form the continuous *backward master equation*

$$\frac{d}{dt}\beta_x(t) = \sum_y q_{xy}\beta_y(t) \ . \tag{3.4}$$

Next, we will calculate the *forward variables* $\mu_x(t)$ and $\gamma_{x,y}(t)$.

Lemma 3.1.10: *For all $t \in [0,T]$ and $x \neq y$ the following equation holds*

$$\gamma_{x,y}(t) = \mu_x(t) q_{x,y} \frac{\beta_y(t)}{\beta_x(t)} \ .$$

Proof: Using the definition of $\gamma_{x,y}(t)$

$$\begin{aligned}
\gamma_{x,y}(t) + \frac{o(h)}{h} &= \frac{\Pr(X^{(t)} = x, X^{(t+h)} = y)}{h} \\
&= \frac{\Pr(X^{(t)} = x|e_0) \cdot \Pr(X^{(t+h)} = y|X^{(t)} = x) \cdot \Pr(e_T|X^{(t+h)} = y)}{h \cdot \Pr(e_T|e_0)} \\
&= \frac{\Pr(X^{(t)} = x|e_0) \cdot q_{x,y} \cdot h \cdot \Pr(e_T|X^{(t+h)} = y)}{h \cdot \Pr(e_T|e_0)} \\
&= \frac{\Pr(X^{(t)} = x|e_0) \cdot \Pr(e_T|X^{(t)} = x)}{\Pr(e_T|e_0)} \cdot q_{x,y} \cdot \frac{\Pr(e_T|X^{(t+h)} = y)}{\Pr(e_T|X^{(t)} = x)} \\
&= \mu_x(t) q_{x,y} \frac{\beta_y(t+h)}{\beta_x(t)}
\end{aligned}$$

and when $h \to 0$ we get the proof. ▮

After calculating these values for each $t \in [0,T]$, using numerical methods (see Appendix A.3) we can now go forward and find the exact marginals $\mu_x(t)$ using the *forward master equation*

$$\frac{d}{dt}\mu_x(t) = \sum_y \gamma_{x,y}(t) \ .$$

This system of ODEs is similar to forward-backward propagation, except that unlike classical forward propagation (which would use a function such as $\alpha_x(t) = \Pr(X^{(t)} = x|\boldsymbol{e}_0)$), here the forward propagation already takes into account the backward messages, to directly compute the posterior. Due to Lemma 3.1.10, we may use the alternative update equation

$$\frac{d}{dt}\mu_x(t) = \sum_y \mu_x(t) q_{x,y} \frac{\beta_y(t)}{\beta_x(t)} \ . \tag{3.5}$$

This equivalence will be made clearer in the context of the optimization procedure that is to come. ▮

The forward-backward scheme described in Example 3.1.9 is not possible for a multi component system, as the number of states is exponential. Therefore, we must resort to a factorized representation, and a different optimization scheme that does not require enumeration over all possible states.

3.2 Factored Approximation

The variational principle we discussed is based on a representation that is as complex as the original process—the number of functions $\gamma_{\boldsymbol{x},\boldsymbol{y}}(t)$ we consider is equal to the size of the original rate matrix $\mathbb{Q}$. To get a tractable inference procedure we make additional simplifying assumptions on the approximating distribution.

Given a D-component process we consider approximations that factor into products of independent processes. More precisely, we define $\mathcal{M}^i_{\boldsymbol{e}}$ to be the continuous Markov-consistent density sets over the component X_i, that are consistent with the evidence on X_i at times 0 and T. Given a collection of density sets $\eta^1, \ldots, \eta^D$ for the different components, the product density set $\eta = \eta^1 \times \cdots \times \eta^D$ is defined as

$$\begin{aligned} \mu_{\boldsymbol{x}}(t) &= \prod_i \mu^i_{x_i}(t) \\ \gamma_{\boldsymbol{x},\boldsymbol{y}}(t) &= \begin{cases} \gamma^i_{x_i,y_i}(t)\mu^{\backslash i}_{\boldsymbol{x}}(t) & \delta(\boldsymbol{x},\boldsymbol{y}) = \{i\} \\ \sum_i \gamma^i_{x_i,x_i}(t)\mu^{\backslash i}_{\boldsymbol{x}}(t) & \boldsymbol{x} = \boldsymbol{y} \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

where $\mu^{\backslash i}_{\boldsymbol{x}}(t) = \prod_{j \neq i} \mu^j_{x_j}(t)$ is the joint distribution at time t of all the components other than the i'th. (It is not hard to see that if $\eta^i \in \mathcal{M}^i_{\boldsymbol{e}}$ for all i, then $\eta \in \mathcal{M}_{\boldsymbol{e}}$). This representation is motivated by the fact that the marginals should factor according to the independent distributions. Defining $\boldsymbol{X} = (x_i)^D_{x=1}$ gives us the factorized distribution

$$\Pr(\boldsymbol{X}(t) = \boldsymbol{x}) = \prod_{i=1}^D \Pr(X_i(t) = x_i) \ .$$

The joint densities are similarly decomposed: denoting $\boldsymbol{y}$ as the vector equal to $\boldsymbol{x}$ everywhere but in the j^{th} component where $\boldsymbol{Y}_j = y_j$ we can write these densities as

$$\Pr(\boldsymbol{X}(t) = \boldsymbol{x}, \boldsymbol{X}(t+h) = \boldsymbol{y}) = \Pr(X_i(t) = x_i, X_i(t+h) = y_i) \cdot \prod_{j \neq i} \Pr(X_j(t) = x_j) \ .$$

We define the set $\mathcal{M}^F_{\boldsymbol{e}}$ to contain all factored density sets. From now on we assume that $\eta = \eta^1 \times \cdots \times \eta^D \in \mathcal{M}^F_{\boldsymbol{e}}$.

Assuming that $\mathbb{Q}$ is defined by a CTBN, and that η is a factored density set, we can use a similar technique used in the Dynamic Bayesian networks Mean Field approximation to rewrite the terms of the average energy functional as a sum factored by each component

$$\begin{aligned} \mathcal{E}(\eta; \mathbb{Q}) = &\sum_i \int_0^T \sum_{x_i} \mu^i_{x_i}(t) \mathbf{E}_{\mu^{\backslash i}(t)} \left[q_{x_i,x_i|\boldsymbol{U}_i} \right] dt \\ &+ \sum_i \int_0^T \sum_{x_i, y_i \neq x_i} \gamma^i_{x_i,y_i}(t) \mathbf{E}_{\mu^{\backslash i}(t)} \left[\ln q_{x_i,y_i|\boldsymbol{U}_i} \right] dt \ . \end{aligned}$$

Proof: We begin with the definition of the average energy

$$\begin{aligned}\mathcal{E}(\eta;\mathbb{Q}) &= \int_0^T \sum_{\boldsymbol{x}} \left[\mu_{\boldsymbol{x}}(t) q_{\boldsymbol{x},\boldsymbol{x}} + \sum_{\boldsymbol{y}\neq\boldsymbol{x}} \gamma_{\boldsymbol{x},\boldsymbol{y}}(t) \ln q_{\boldsymbol{x},\boldsymbol{y}} \right] dt \\ &= \int_0^T \sum_{\boldsymbol{x}} \left[\mu_{\boldsymbol{x}}(t) q_{\boldsymbol{x},\boldsymbol{x}} + \sum_{i} \sum_{y_i\neq x_i} \gamma^i_{x_i,y_i}(t) \mu^{\setminus i}(t) \ln q_{\boldsymbol{x},\boldsymbol{y}} \right] dt\end{aligned}$$

where the equality stems from the observation that the only states $\boldsymbol{y}$ that may have $\gamma_{\boldsymbol{x},\boldsymbol{y}}(t) > 0$, are those with $\delta(\boldsymbol{x},\boldsymbol{y}) \leq 1$ (all the rest are 0). Thus, the enumeration over all possible states collapses into an enumeration over all components i and all states $y_i \neq x_i$. Due to the fact that we are only considering transitions in single components, we may replace the global joint density $\gamma_{\boldsymbol{x},\boldsymbol{y}}$ with $\gamma^i_{x_i,y_i} \cdot \mu^{\setminus i}(t)$, as per definition.

Using (2.10), we can decompose the transition rates $q_{\boldsymbol{x},\boldsymbol{x}}$ and $q_{\boldsymbol{x},\boldsymbol{y}}$ to get

$$\begin{aligned}&\int_0^T \sum_{\boldsymbol{x}} \left[\mu_{\boldsymbol{x}}(t) q_{\boldsymbol{x},\boldsymbol{x}} + \sum_{i} \sum_{y_i\neq x_i} \gamma^i_{x_i,y_i}(t) \mu^{\setminus i}(t) \ln q_{\boldsymbol{x},\boldsymbol{y}} \right] dt \\ &= \sum_i \int_0^T \sum_{\boldsymbol{x}} \left[\mu_{\boldsymbol{x}}(t) q_{x_i,x_i|\boldsymbol{u}_i} + \sum_{y_i\neq x_i} \gamma^i_{x_i,y_i}(t) \mu^{\setminus i}(t) \ln q_{x_i,y_i|\boldsymbol{u}_i} \right] dt \\ &= \sum_i \int_0^T \sum_{x_i} \left[\mu^i_{x_i}(t) \sum_{\boldsymbol{x}\setminus i} \mu^{\setminus i}_{\boldsymbol{x}\setminus i}(t) q_{x_i,x_i|\boldsymbol{u}_i} + \sum_{y_i\neq x_i} \gamma^i_{x_i,y_i}(t) \mu^{\setminus i}_{\boldsymbol{x}\setminus i}(t) \ln q_{x_i,y_i|\boldsymbol{u}_i} \right] dt \ .\end{aligned}$$

To get to the last equality we use the factorization of $\mu(t)$ as a product of $\mu^i(t)$ with $\mu^{\setminus i}(t)$ and the reordering of the summation. Next we simply write the previous sum as an expectation over $\boldsymbol{X} \setminus i$

$$\begin{aligned}&\sum_i \int_0^T \sum_{x_i} \left[\mu^i_{x_i}(t) \sum_{\boldsymbol{x}\setminus i} \mu^{\setminus i}_{\boldsymbol{x}\setminus i}(t) q_{x_i,x_i|\boldsymbol{u}_i} + \sum_{y_i\neq x_i} \gamma^i_{x_i,y_i}(t) \mu^{\setminus i}_{\boldsymbol{x}\setminus i}(t) \ln q_{x_i,y_i|\boldsymbol{u}_i} \right] dt \\ =\ &\sum_i \int_0^T \sum_{x_i} \mu^i_{x_i}(t) \mathbf{E}_{\mu^{\setminus i}(t)} \left[q_{x_i,x_i|\boldsymbol{U}_i} \right] + \sum_i \int_0^T \sum_{y_i\neq x_i} \gamma^i_{x_i,y_i}(t) \mathbf{E}_{\mu^{\setminus i}(t)} \left[\ln q_{x_i,y_i|\boldsymbol{U}_i} \right] dt\end{aligned}$$

which concludes the proof. ▮

A further simplification for the calculation of the average energy can be found when observing the expectations over the transition rates $\mathbf{E}_{\mu^{\setminus i}(t)} \left[q_{x_i,y_i|\boldsymbol{U}_i} \right]$. As the rates are independent of all the other components given the state of $\mathbf{Pa}_i$, it is sufficient to sum over all states of $\mathbf{Pa}_i$

$$\mathbf{E}_{\mu^{\setminus i}(t)} \left[q_{x_i,y_i|\boldsymbol{U}_i} \right] = \mathbf{E}_{\mu^{\mathbf{Pa}_i}(t)} \left[q_{x_i,y_i|\boldsymbol{U}_i} \right]$$

because the left hand side involves only $\mu^j(t)$ for $j \in \mathbf{Pa}_i$. Similarly, the entropy-like term factorizes as

$$\mathcal{H}(\eta) = \sum_i \mathcal{H}(\eta^i).$$

This decomposition involves only local terms that either include the i'th component, or include the i'th component and its parents in the CTBN defining $\mathbb{Q}$. Thus, this decomposition allows us to write $\mathcal{F}(\eta;\mathbb{Q})$ as a function of $\mu^i_{x_i}$ and $\gamma^i_{x_i,y_i}$ for $x_i \neq y_i$. To make the factored nature of the approximation explicit in the notation, we write henceforth,

$$\mathcal{F}(\eta;\mathbb{Q}) = \tilde{\mathcal{F}}(\eta^1,\ldots,\eta^D;\mathbb{Q}).$$

3.3 Optimization

Our goal is to maximize the functional, but to keep the η^i Markov consistent densities. Thus, the maximization of the factored energy functional demands the use of *constrained optimization* techniques in the field of *Lagrange multipliers*, elicited in Appendix A.1. Due to the independence between the different η^i, we will perform *block ascent*, optimizing the functional with respect to each parameter set in turn.

Fixed Point Characterization We can now pose the optimization problem we wish to solve:

> Fixing i, and given $\eta^1,\ldots,\eta^{i-1},\eta^{i+1},\ldots,\eta^D$, in $\mathcal{M}^1_{\boldsymbol{e}},\ldots\mathcal{M}^{i-1}_{\boldsymbol{e}},\mathcal{M}^{i+1}_{\boldsymbol{e}},\ldots,\mathcal{M}^D_{\boldsymbol{e}}$, respectively, find
> $$\arg\max_{\eta^i\in\mathcal{M}^i_{\boldsymbol{e}}} \tilde{\mathcal{F}}(\eta^1,\ldots,\eta^D;\mathbb{Q}) \ .$$

If for all i, we have a $\mu^i \in \mathcal{M}^i_{\boldsymbol{e}}$, which is a solution to this optimization problem with respect to each component, then we have a (local) stationary point of the energy functional within $\mathcal{M}^F_{\boldsymbol{e}}$. Additionally, due to Lemma 3.1.4, each η which is Markov consistent defines a valid Markov process, and thus we will always remain in the domain of consistent parameterization.

To solve this optimization problem, we define a Lagrangian, which includes the constraints in the form of Def. 3.1.3. These constraints are to be enforced in a continuous fashion, and so the Lagrange multipliers $\lambda^i_{x_i}(t)$ are continuous functions of t as well. The Lagrangian is a functional of the functions $\mu^i_{x_i}(t)$, $\gamma^i_{x_i,y_i}(t)$ and $\lambda^i_{x_i}(t)$, and takes the following form

$$\mathcal{L} = \tilde{\mathcal{F}}(\eta;\mathbb{Q}) - \sum_{i=1}^{D}\int_0^T \lambda^i_{x_i}(t)\left(\frac{d}{dt}\mu^i_{x_i}(t) - \sum_{y_i}\gamma^i_{x_iy_i}(t)\right)dt \ .$$

Similarly to the discrete case, we need to find the derivatives of the Lagrangian with respect to each variable and form fixed point equations which will show us how to update the parameters. Despite this similarity, due to its continuous nature, our derivations are now w.r.t. *continuous functions*, not discrete parameters. This requires the use of so-called *functional derivatives* and several techniques in *Calculus of variations*. The stationary point of the Lagrangian satisfies the *Euler-Lagrange* equations (see Appendix A.1). Writing these equations in explicit form, we get a fixed point characterization of the solution in term of the following set of ODEs:

$$\begin{aligned}\frac{d}{dt}\mu^i_{x_i}(t) &= \sum_{y_i \neq x_i} \left(\gamma^i_{y_i,x_i}(t) - \gamma^i_{x_i,y_i}(t)\right) \\ \frac{d}{dt}\rho^i_{x_i}(t) &= -\rho^i_{x_i}(t)(\bar{q}^i_{x_i,x_i}(t) + \psi^i_{x_i}(t)) - \sum_{y_i \neq x_i} \rho^i_{y_i}(t)\tilde{q}^i_{x_i,y_i}(t)\end{aligned} \tag{3.6}$$

where ρ^i are the exponents of the Lagrange multipliers λ_i as defined in the proof of Thm. 3.3.2 below. In addition we get the following algebraic constraint

$$\rho^i_{x_i}(t)\gamma^i_{x_i,y_i}(t) = \mu^i_{x_i}(t)\tilde{q}^i_{x_i,y_i}(t)\rho^i_{y_i}(t), \;\; x_i \neq y_i. \tag{3.7}$$

In these equations we use the following shorthand notations for the average rates

$$\begin{aligned}\bar{q}^i_{x_i,y_i}(t) &= \mathbf{E}_{\mu^{\setminus i}(t)}\left[q^{i|\mathbf{Pa}_i}_{x_i,y_i|\boldsymbol{U}_i}\right] \\ \bar{q}^i_{x_i,y_i|x_j}(t) &= \mathbf{E}_{\mu^{\setminus i}(t)}\left[q^{i|\mathbf{Pa}_i}_{x_i,y_i|\boldsymbol{U}_i} \mid x_j\right],\end{aligned}$$

where $\mu^{\setminus i}(t)$ is the product distribution of $\mu^1(t), \ldots, \mu^{i-1}(t), \mu^{i+1}(t), \ldots, \mu^D(t)$. Similarly, we have the following shorthand notations for the geometrically-averaged rates,

$$\begin{aligned}\tilde{q}^i_{x_i,y_i}(t) &= \exp\left\{\mathbf{E}_{\mu^{\setminus i}(t)}\left[\ln q^{i|\mathbf{Pa}_i}_{x_i,y_i|\boldsymbol{U}_i}\right]\right\} \\ \tilde{q}^i_{x_i,y_i|x_j}(t) &= \exp\left\{\mathbf{E}_{\mu^{\setminus i}(t)}\left[\ln q^{i|\mathbf{Pa}_i}_{x_i,y_i|\boldsymbol{U}_i} \mid x_j\right]\right\} \quad .\end{aligned}$$

The last auxiliary term is

$$\begin{aligned}\psi^i_{x_i}(t) = &\sum_{j \in \mathit{Children}_i} \sum_{x_j} \mu^j_{x_j}(t)\bar{q}^j_{x_j,x_j|x_i}(t) + \\ &\sum_{j \in \mathit{Children}_i} \sum_{x_j \neq y_j} \gamma^j_{x_j,y_j}(t) \ln \tilde{q}^j_{x_j,y_j|x_i}(t) \quad .\end{aligned}$$

The two differential equations (3.6) for $\mu^i_{x_i}(t)$ and $\rho^i_{x_i}(t)$ describe, respectively, the progression of $\mu^i_{x_i}$ forward, and the progression of $\rho^i_{x_i}$ backward. To uniquely solve these equations we need to set the boundary conditions. The boundary condition for $\mu^i_{x_i}$ is defined explicitly in $\mathcal{M}^F_{\boldsymbol{e}}$ as

$$\mu^i_{x_i}(0) = \delta_{x_i,e_{i,0}} \tag{3.8}$$

The boundary condition at T is slightly more involved. The constraints in $\mathcal{M}^F_{\boldsymbol{e}}$ imply that $\mu^i_{x_i}(T) = \delta_{x_i,e_{i,T}}$. These constraints that $\mu_{\boldsymbol{x}}(0)$ and $\mu_{\boldsymbol{x}}(T)$ are extreme probabilities (having all the mass on one value) have consequences on $\gamma_{\boldsymbol{x},\boldsymbol{y}}$ at these points, as stated the following lemma

Lemma 3.3.1: *If $\eta \in \mathcal{M}_{\boldsymbol{e}}$ then $\gamma_{\boldsymbol{x},\boldsymbol{y}}(0) = 0$ for all $\boldsymbol{x} \neq \boldsymbol{e}_0$ and $\gamma_{\boldsymbol{x},\boldsymbol{y}}(T) = 0$ for all $\boldsymbol{y} \neq \boldsymbol{e}_T$.*

Proof: If $\boldsymbol{y} \neq \boldsymbol{e}_T$ then $P_\eta(\boldsymbol{X}^{(T-h)} = \boldsymbol{x}, \boldsymbol{X}^{(T)} = \boldsymbol{y}) = 0$, and thus by (3.1) and the continuity of $\gamma_{\boldsymbol{x},\boldsymbol{y}}$ we have that $\gamma_{\boldsymbol{x},\boldsymbol{y}}(0) = 0$. ∎

As stated by Lemma 3.3.1, we have that $\gamma^i_{e_{i,T},x_i}(T) = 0$ when $x_i \neq e_{i,T}$. Plugging these values into (3.7), and assuming that $\mathbb{Q}$ is irreducible we get that $\rho_{x_i}(T) = 0$ for all $x_i \neq e_{i,T}$. In addition, we notice that $\rho_{e_{i,T}}(T) \neq 0$, for otherwise the whole system of equations for ρ will collapse to 0. Finally, notice that the solution of (3.6) for μ^i and γ^i is insensitive to the multiplication of ρ^i by a constant. Thus, we can arbitrarily set $\rho_{e_{i,T}}(T) = 1$, and get the boundary condition

$$\rho^i_{x_i}(T) = \delta_{x_i, e_{i,T}}. \tag{3.9}$$

We can now prove the correctness of our sequential updates in the algorithm portrayed in 3.3 that uses (3.6–3.9)

Theorem 3.3.2: $\eta^i \in \mathcal{M}^i_{\boldsymbol{e}}$ is a stationary point (e.g., local maxima) of $\tilde{\mathcal{F}}(\eta^1, \ldots, \eta^D; \mathbb{Q})$ subject to the constraints of Def. 3.1.3 if and only if it satisfies (3.6–3.9).

Proof: The Euler-Lagrange equations of the Lagrangian define its stationary points w.r.t. the parameters of each component $\mu^i(t)$, $\gamma^i(t)$ and $\lambda^i(t)$. First, taking derivatives w.r.t. $\mu^i_{x_i}(t)$ gives us the following equality

$$\bar{q}^i_{x_i,x_i}(t) - \frac{\gamma^i_{x_i,x_i}}{\mu^i_{x_i}(t)} + \frac{d}{dt}\lambda^i_{x_i}(t) + \psi^i_{x_i}(t) = 0 \ . \tag{3.10}$$

The derivative of $\lambda^i_{x_i}(t)$ term is obtained from the time-derivative of $\mu^i_{x_i}(t)$ in the Lagrangian, as described in (A.1), and the $\psi^i_{x_i}(t)$ term is derived from the equality

$$\frac{\partial}{\partial \mu^i_{x_i}(t)} \sum_{j \neq i} \sum_{x_j} \mu^i_{x_i}(t) \mathbf{E}_{\mu^{\setminus i}(t)} \left[q^i_{x_i,x_i|\boldsymbol{U}_i} \right] = \psi^i_{x_i}(t) \ .$$

Next, the derivative w.r.t. $\gamma^i_{x_i,y_i}(t)$ gives us

$$\ln \mu^i_{x_i}(t) + \ln \tilde{q}^i_{x_i,y_i}(t) - \ln \gamma^i_{x_i,y_i}(t) + \lambda^i_{y_i}(t) - \lambda^i_{x_i}(t) = 0 \ . \tag{3.11}$$

Finally, the derivative w.r.t. $\lambda^i_{x_i}(t)$ gives us the update equation of the $\mu^i(t)$, so it is Markov consistent

$$\mu^i_{x_i}(t) - \sum_{y_i \neq x_i} \left[\gamma^i_{y_i,x_i}(t) - \gamma_{x_i,y_i}(t) \right] = 0 \ .$$

Denoting $\rho^i_{x_i}(t) = \exp\{\lambda^i_{x_i}(t)\}$ we have the relation

$$\frac{d}{dt}\rho^i_{x_i}(t) = \rho^i_{x_i}(t) \cdot \frac{d}{dt}\lambda^i_{x_i}(t)$$

which transforms (3.11) into

$$\gamma^i_{x_i,y_i}(t) = \mu^i_{x_i}(t) \bar{q}^i_{x_i,y_i}(t) \frac{\rho^i_{y_i}(t)}{\rho^i_{x_i}(t)} \ .$$

Thus by definition

$$\gamma^i_{x_i,x_i}(t) = -\sum_{y_i \neq x_i} \gamma^i_{x_i,y_i}(t) = -\mu^i_{x_i}(t) \sum_{x_i,y_i} \tilde{q}^i_{x_i,y_i}(t) \frac{\rho^i_{y_i}(t)}{\rho^i_{x_i}(t)}.$$

Plugging this equality into (3.10) and using the fact that

$$\frac{d}{dt}\rho^i_{x_i}(t) = \frac{d}{dt}\lambda^i_{x_i}(t)\rho^i_{x_i}(t)$$

gives us

$$\frac{d}{dt}\rho^i_{x_i}(t) = -\rho^i_{x_i}(t)\left(\bar{q}^i_{x_i,x_i|\boldsymbol{U}_i}(t) + \psi^i_{x_i}(t)\right) - \sum_{y_i \neq x_i} \tilde{q}^i_{x_i,y_i}\rho^i_{y_i}(t) \ .$$

Thus the stationary point of the lagrangian matches the updates of (3.6–3.7). ▮

It is straightforward to extend this result to show that at a maximum with respect to all the component densities, this fixed-point characterization must hold for all components simultaneously.

Example 3.3.3: Returning to Example 3.1.9, the averaged rates $\bar{q}^i$ and the geometrically-averaged rates $\tilde{q}^i$ both reduce to the unaveraged rates q, and $\psi \equiv 0$. Thus, the system of equations to be solved is

$$\begin{aligned}
\frac{d}{dt}\mu_x(t) &= \sum_{y \neq x}(\gamma_{y,x}(t) - \gamma_{x,y}(t)) \\
\frac{d}{dt}\rho_x(t) &= -\sum_{y} q_{x,y}\rho_y(t),
\end{aligned}$$

along with the algebraic equation

$$\rho_x(t)\gamma_{x,y}(t) = q_{x,y}\mu_x(t)\rho_y(t), \qquad y \neq x.$$

In this case, it is straightforward to show that the backward propagation rule for ρ_x implies that

$$\rho_x(t) = \Pr(e_T|X^{(t)})$$

as for every x, $\rho_x(T) = \delta_{e_T,x}$ and its backward propagation rule is exactly the backward master equation from (3.4). Setting the marginals at $t = 0$ to be consistent with the evidence, the forward propagation rule

$$\frac{d}{dt}\mu_x(t) = \mu_x(t)\sum_{y} q_{x,y}\frac{\rho_y(t)}{\rho_x(t)}$$

exactly matches the one in Lemma 3.1.10, and thus correctly calculates the marginals.

This interpretation of $\rho_x(t)$ also allows us to understand the role of $\gamma_{x,y}(t)$. Recall that $\gamma_{x,y}(t)/\mu_x(t)$ is the instantaneous rate of transition from x to y at time t. Thus,

$$\frac{\gamma_{x,y}(t)}{\mu_x(t)} = q_{x,y}\frac{\rho_y(t)}{\rho_x(t)}.$$

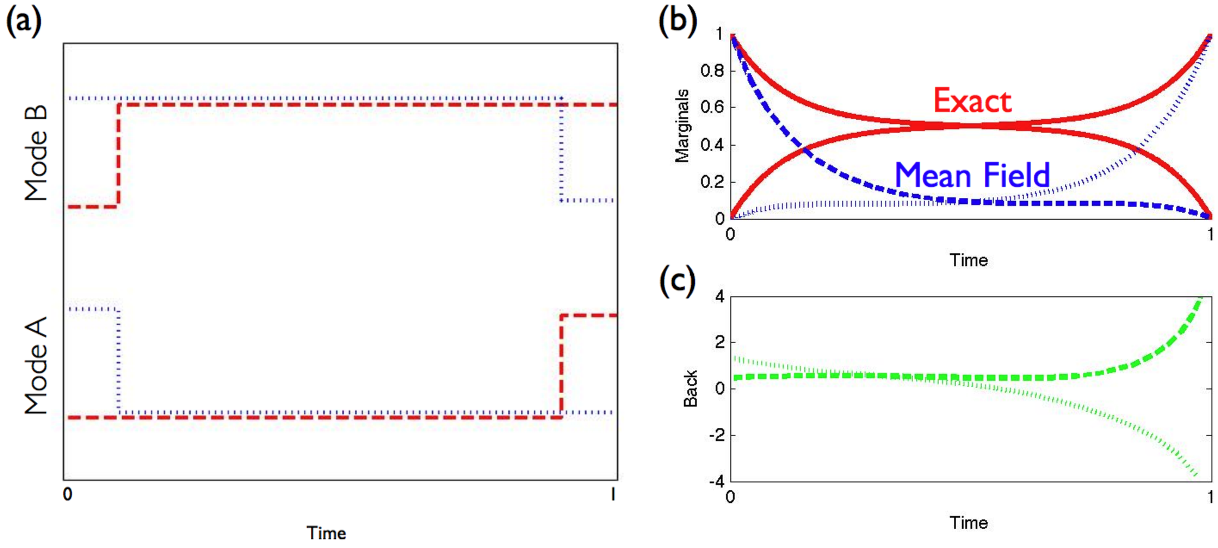

Figure 3.1: Numerical results for the two-component Ising chain described in Example 3.3.4 where the first component starts in state -1 and ends at time $T = 1$ in state 1. The second component has the opposite behavior. **(a)** Two likely trajectories depicting the two modes of the model. **(b)** Exact (solid) and approximate (dashed/dotted) marginals $\mu^i_1(t)$. **(c)** The log ratio $\log \rho^i_1(t)/\rho^i_0(t)$.

That is, the instantaneous rate combines the original rate with the relative likelihood of the evidence at T given y and x. If y is much more likely to lead to the final state, then the rates are biased toward y. Conversely, if y is unlikely to lead to the evidence the rate of transitions to it are lower. This observation also explains why the forward propagation of μ_x will reach the observed $\mu_x(T)$ even though we did not impose it explicitly. ▮

An example of a multicomponent process is the *Ising chain*, which corresponds to a CTBN $X_1 \leftrightarrow \cdots \leftrightarrow X_D$ such that each binary component prefers to be in the same state as its neighbour. These models are governed by two parameters: a *coupling parameter* β (it is the inverse temperature in physical models, which determines the strength of the coupling between two neighboring components, and a *rate parameter* τ that determines the propensity of each component to change its state. Low values of β correspond to weak coupling (high temperature). More formally, we define the conditional rate matrices as

$$q^{i|\mathbf{Pa}_i}_{x_i,y_i|\boldsymbol{u}_i} = \tau\left(1 + e^{-2y_i\beta\sum_{j\in\mathbf{Pa}_i} x_j}\right)^{-1}$$

where $x_j \in \{-1, 1\}$. This model is derived from the Ising grid on *Continuous-Time Markov Networks*, which are the undirected counterparts of CTBNs, by using the technique in El-Hay et al. (2006).

Example 3.3.4: Let us return to the two-component Ising chain in Example 2.3.1 with initial state $X^{(0)}_1 = -1$ and $X^{(0)}_2 = 1$, and a reversed state at the final time, $X^{(T)}_1 = 1$ and $X^{(T)}_2 = -1$. For a

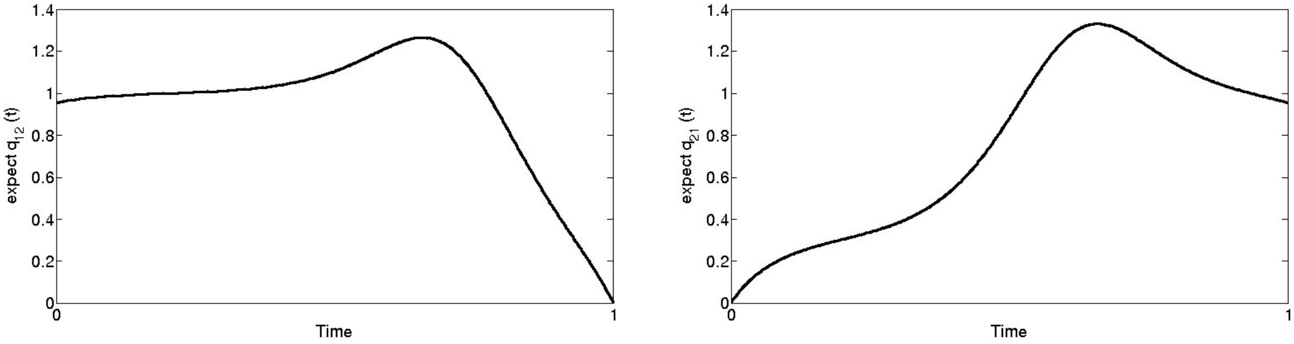

Figure 3.2: The expected rates $\vec{q}^i(t)$ of the approximating process in each of the components of the (symmetric) Ising chain in Example 2.3.1. In the left figure, we can see transitions $\vec{q}^i_{1,2}(t)$ and in the right $\vec{q}^i_{2,1}(t)$. We can notice that the averaged rates are highly non-linear, and so are cannot be approximated well with a constant rate matrix.

large value of β, this evidence is unlikely as at both end points the components are in a undesired configurations. The exact posterior is one that assigns higher probabilities to trajectories where one of the components switches relatively fast to match the other, and then toward the end of the interval, they separate to match the evidence. Since the model is symmetric, these trajectories are either ones in which both components are most of the time in state -1, or ones where both are most of the time in state 1 (Fig. 3.1 (a)). Due to symmetry, the marginal probability of each component is around 0.5 throughout most of the interval (Fig. 3.1 (b)). The variational approximation cannot capture the dependency between the two components, and thus converges to one of two local maxima, corresponding to the two potential subsets of trajectories. Examining the value of ρ^i, we see that close to the end of the interval they bias the instantaneous rates significantly (Fig. 3.1 (c)).

This example also allows to examine the implications of modeling the posterior by inhomogeneous Markov processes. In principle, we might have used as an approximation Markov processes with homogeneous rates, and conditioned on the evidence. To examine whether our approximation behaves in this manner, we notice that in the single component case we have

$$q_{x,y} = \frac{\rho_x(t)\gamma_{x,y}(t)}{\rho_y(t)\mu_x(t)},$$

which should be constant.

Consider the analogous quantity in the multi-component case: $\tilde{q}^i_{x_i,y_i}(t)$, the geometric average of the rate of X_i, given the probability of parents state. Not surprisingly, this is exactly a mean field approximation, where the influence of interacting components is approximated by their average influence. Since the distribution of the parents (in the two-component system, the other component) changes in time, these rates change continuously, especially near the end of the time interval. This suggests that a piecewise homogeneous approximation cannot capture the dynamics without a loss in accuracy. As expected in a dynamic process, we can see in Fig. 3.2 that the inhomogeneous transition rates are very erratic. In particular, the rates spikes at the coordinated point of transition

```
For each $i$, initialize $\mu^i$ using $\mathbb{Q}^{i|\boldsymbol{u}_i}$ with some random state $\boldsymbol{u}_i$.
while not converged do
    1. Pick a component $i \in \{1, \ldots, D\}$.
    2. Update $\rho^i(t)$ by solving the $\rho^i$ backward master equation in (3.6).
    3. Update $\mu^i(t)$ and $\gamma^i(t)$ by solving the $\mu^i$ forward master equation in (3.6) and the fixed
       equation in (3.7).
end
```

Figure 3.3: Mean Field approximation in continuous-time Bayesian networks

selected by the Mean Field approximation. This can be interpreted as putting all the weight of the distribution on trajectories which transition from state -1 to 1 at that point. ▮

Optimization Procedure If $\mathbb{Q}$ is irreducible, then $\rho^i_{x_i}$ and μ_{x_i} are non-zero throughout the open interval $(0, T)$. As a result, we can solve (3.7) to express $\gamma^i_{x_i,y_i}$ as a function of μ^i and ρ^i, thus eliminating it from (3.6) to get evolution equations solely in terms of μ^i and ρ^i. Abstracting the details, we obtain a set of ODEs of the form

$$\begin{aligned} \frac{d}{dt}\mu^i(t) &= \alpha(\mu^i(t), \rho^i(t), \mu^{\setminus i}(t)) \quad && \mu^i(0) = \text{given} \\ \frac{d}{dt}\rho^i(t) &= -\beta(\rho^i(t), \mu^{\setminus i}(t)) && \rho^i(T) = \text{given.} \end{aligned}$$

where α and β can be inferred from (3.6) and (3.7) The general optimization procedure can be seen in 3.3. First we initialize all the marginals and joint probabilities of each component i according to a conditional rate matrix $\mathbb{Q}^{i|\boldsymbol{u}_i}$ using some random assignment $\boldsymbol{u}_i$. Next we iterate in updating each component according to (3.7). First we solve $\gamma^i(t)$ backwards, then we update $\gamma^i(t)$ and $margi(t)$ forward using the forward master equation. We can measure the improvement of the approximation by calculating the functional after the update and comparing it to its previous value. We stop once we have reached a point where the functional converges.

Since the evolution of ρ^i does not depend on μ^i, we can integrate backward from time T to solve for ρ^i. Then, integrating forward from time 0, we compute μ^i. After performing a single iteration of backward-forward integration, we obtain a solution that satisfies the fixed-point equation (3.6) for the i'th component. (This is not surprising once we have identified our procedure to be a variation of a standard forward-backward algorithm for a single component.) Such a solution will be a local maximum of the functional w.r.t. to η^i (reaching a local minimum or a saddle point requires very specific initialization points).

This suggests that we can use the standard procedure of asynchronous updates, where we update each component in a round-robin fashion. Since each of these single-component updates

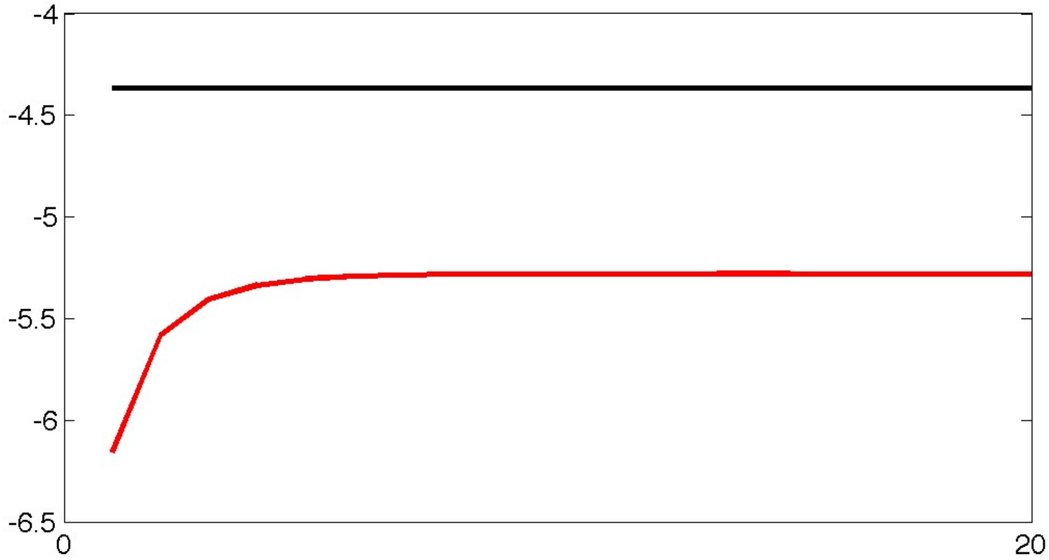

Figure 3.4: Convergence of the energy functional in a 8 component ising chain in 20 iterations. The functional (in red) is a lower bound on the log likelihood of the (black), and increases monotonically with each iteration.

converges in one backward-forward step, Wand since it reaches a local maximum, each step improves the value of the free energy over the previous one (Fig. 3.4). Since the free energy functional is bounded by the probability of the evidence, this procedure will always converge.

Potentially, there can be many scheduling possibilities. In our implementation the update scheduling is simply random. A better choice would be to update the component which would maximally increase the value of the functional in that iteration. This idea is similar the scheduling of Elidan et al. (2006), who approximate the change in the beliefs by bounding the *residuals* of the messages, which give an approximation of the benefit of updating each component.

Another issue is the initialization of this procedure. Since the iteration on the i'th component depends on $\mu^{\setminus i}$, we need to initialize μ by some legal assignment. To do so, we create a fictional rate matrix $\tilde{\mathbb{Q}}_i$ for each component and initialize μ^i to be the posterior of the process given the evidence $e_{i,0}$ and $e_{i,T}$. As a reasonable initial guess, we choose at random one of the conditional rates in $\mathbb{Q}$ to determine the fictional rate matrix.

Exploiting the Continuous Time Representation The continuous time update equations allow us to use standard ODE methods with an adaptive step size (here we use the Runge-Kutta-Fehlberg (4,5) method). At the price of some overhead, these procedures automatically tune the tradeoff between error and time granularity. See Appendix A.3 for more information on numerical integration.

To further save computations, we note that while standard integration methods involve only

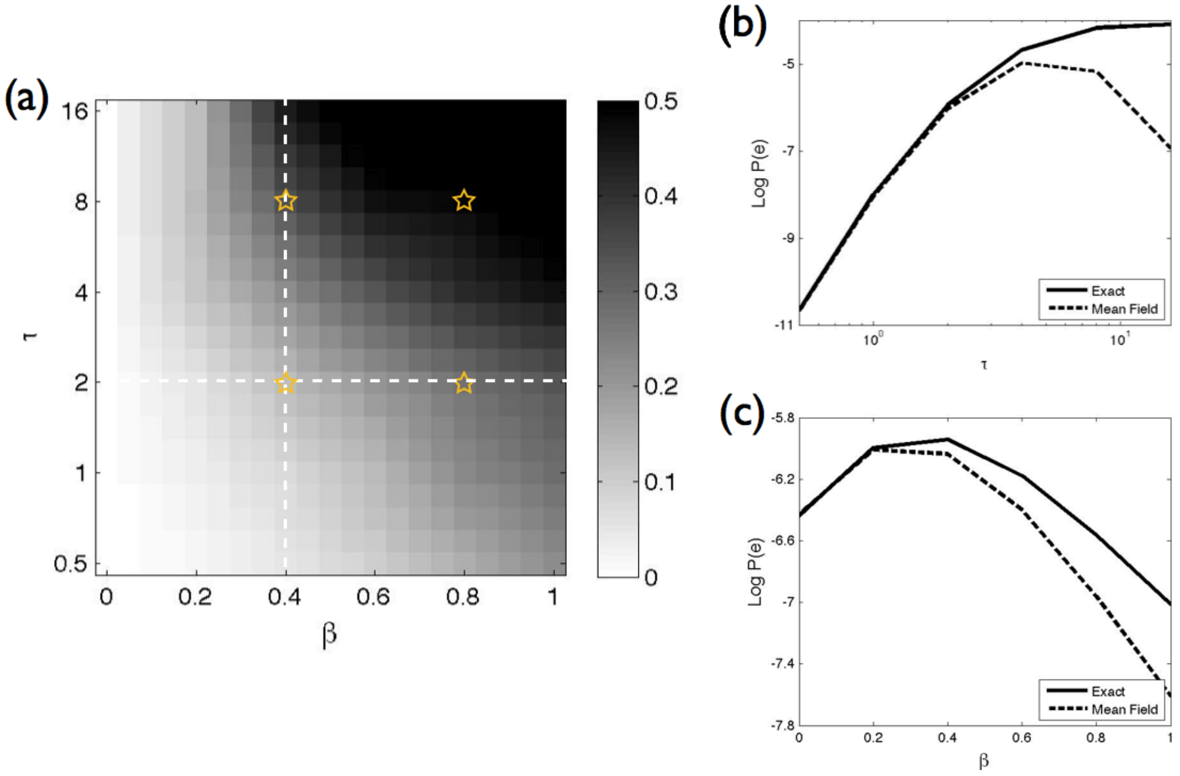

Figure 3.5: **(a)** Relative error as a function of the coupling parameter β (x-axis) and transition rates τ (y-axis) for an 8-component Ising chain. **(b)** Comparison of true vs. estimated likelihood as a function of the rate parameter τ. **(c)** Comparison of true vs. likelihood as a function of the coupling parameter β.

initial boundary conditions at $t = 0$, the solution of μ^i is also known at $t = T$. Therefore, we stop the adaptive integration when $\mu^i(t) \approx \mu^i(T)$ and t is close enough to T. This modification reduces the number of computed points significantly because the derivative of μ^i tends to grow near the boundary due to its exponential behaviour, resulting in a lower step size.

3.4 Evaluation

To gain better insight into the quality of our procedure, we performed numerical tests on models that challenge the approximation. Specifically, we use Ising chains where we explore regimes defined by the degree of coupling between the components (the parameter β) and the rate of transitions (the parameter τ). We evaluate the error in two ways. The first is by the difference between the true log-likelihood and our estimate. The second is by the average relative error in the estimate of different expected sufficient statistics defined by $\sum_j \frac{|\hat{\theta}_j - \theta_j|}{\theta_j}$ where θ_j is exact value of the j'th expected sufficient statistics and $\hat{\theta}_j$ is the approximation.

Applying our procedure on an Ising chain with 8 components, for which we can still perform exact inference, we evaluated the relative error for different choices of β and τ. The evidence in this experiment is $\boldsymbol{e}_0 = \{+, +, +, +, +, +, -, -\}$, $T = 0.64$ and $\boldsymbol{e}_T = \{-, -, -, +, +, +, +, +\}$. As shown in Fig. 3.5a, the error is larger when τ and β are large. In the case of a weak coupling (small β), the posterior is almost independent, and our approximation is accurate. In models with few transitions (small τ), most of the mass of the posterior is concentrated on a few canonical "types" of trajectories that can be captured by the approximation (as in Example 3.3.4). At high transition rates, the components tend to transition often, and in a coordinated manner, which leads to a posterior that is hard to approximate by a product distribution. Moreover, the resulting free energy landscape is rough with many local maxima. Examining the error in likelihood estimates (Fig. 3.5b,c) we see a similar trend. Other than approximating the expected statistics, a good approximation should also find the *maximum-likelihood* model, i.e., the model θ^* which gives $\arg\max_\theta \Pr(\boldsymbol{e}; \theta)$. It can be easily seen that this approximation does not find the max-likelihood model (Fig. 3.5b,c).

Next, we examine the run time of our approximation when using fairly standard ODE solver with few optimizations and tunings. The run time is dominated by the time needed to perform the backward-forward integration when updating a single component, and by the number of such updates until convergence. Examining the run time for different choices of β and τ (Fig. 3.6), we see that the run time of our procedure scales linearly with the number of components in the chain. Moreover, the run time is generally insensitive to the difficulty of the problem in terms of β. It does depend to some extent on the rate τ, suggesting that processes with more transitions require more iterations to converge. Indeed, the number of iterations required to achieve convergence in the largest chains under consideration are mildly affected by parameter choices.

The scalability of the run time stands in contrast to the Gibbs sampling procedure (El-Hay et al., 2008), which scales roughly with the number in transitions in the sampled trajectories. Comparing our method to the Gibbs sampling procedure we see (Fig. 3.7) that the faster Mean Field approach dominates the Gibbs procedure over short run times. However, as opposed to Mean Field, the Gibbs procedure is asymptotically unbiased, and with longer run times it ultimately prevails. This evaluation also shows that the adaptive integration procedure in our methods strikes a better trade-off than using a fixed time granularity integration.

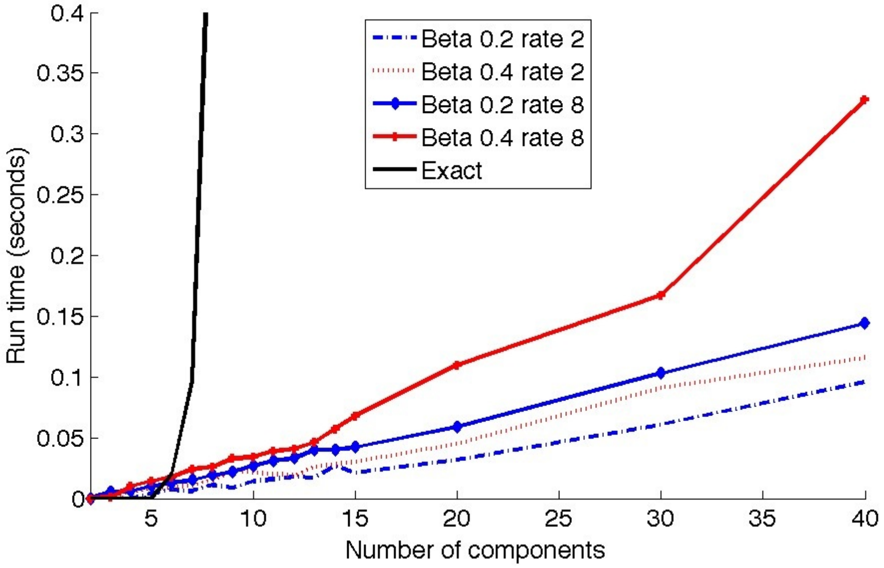

Figure 3.6: Evaluation of the run time of the approximation versus the run time of exact inference as a function of the number of components.

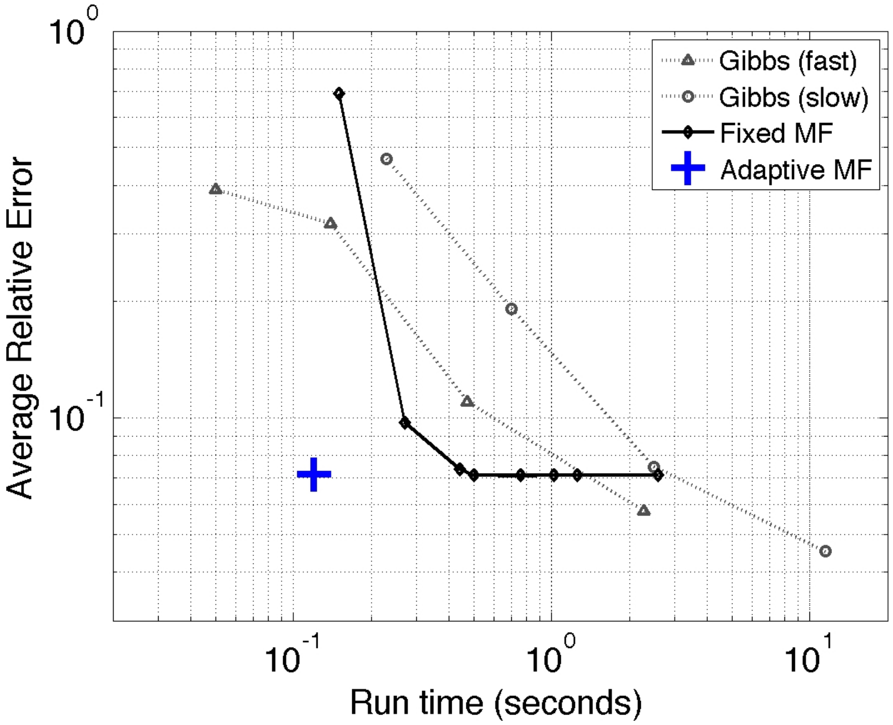

Figure 3.7: Evaluation of the run time vs. accuracy trade-off for several choices of parameters for Mean Field and Gibbs sampling on the branching process of Fig. 4.2.

Chapter 4

Branching Processes

The abovementioned experimental results indicate that our approximation is accurate when reasoning about weakly-coupled components, or about time intervals involving few transitions (low transition rates). Unfortunately, in many domains we face strongly-coupled components. For example, we are interested in modeling the evolution of biological sequences (DNA, RNA, and proteins). In such systems, we have a *phylogenetic tree* that represents the branching process that leads to current day sequences (see Fig. 4.2). It is common in sequence evolution to model this process as a continuous-time Markov process over a tree (Felsenstein, 2004). More precisely, the evolution along each branch is a standard continuous-time Markov process, and branching is modeled by a replication, after which each replica evolves independently along its sub-branch. Common applications are forced to assume that each character in the sequence evolves independently of the other.

In some situations, assuming an independent evolution of each character is highly unreasonable. Consider the evolution of an RNA sequence that folds onto itself to form a functional structure. This folding is mediated by complementary base-pairing (A-U, C-G, etc) that stabilizes the structure. During evolution, we expect to see compensatory mutations. That is, if a A changes into C then its based-paired U will change into a G soon thereafter. To capture such coordinated changes, we need to consider the joint evolution of the different characters. In the case of RNA structure, the stability of the structure is determined by *stacking potentials* that measure the stability of two adjacent pairs of interacting nucleotides. Thus, if we consider a factor network to represent the energy of a fold, it will have structure as shown in Fig. 4.1b. We can convert this factor graph into a CTBN using procedures that consider the energy function as a fitness criteria in evolution (El-Hay et al., 2006; Yu and Thorne, 2006). Unfortunately, inference in such models suffers from computational blowup, and so the few studies that deal with it explicitly resort to sampling procedures (Yu and Thorne, 2006).

4.1 Representation

To consider trees, we need to extend our framework to deal with branching processes. In a linear-time model, we view the process as a map from $[0, T]$ into random variables $\boldsymbol{X}^{(t)}$. In the case of

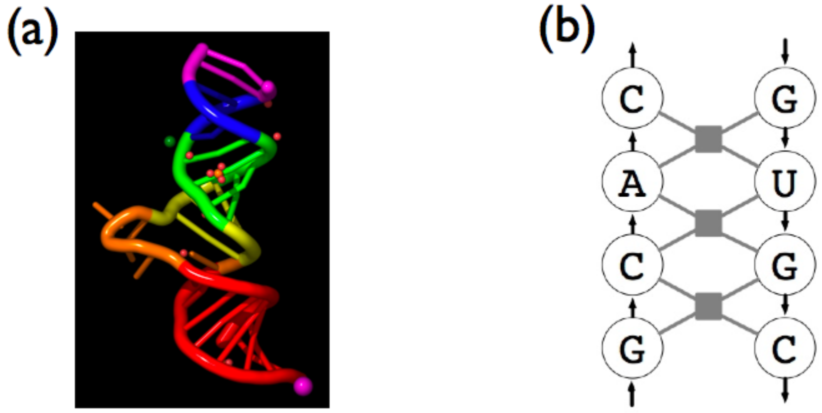

Figure 4.1: (a) Structure of an RNA molecule. The 3 dimensional structure dictates the dependencies between the different positions. **(b)** The form of the energy function for encoding RNA folding, superimposed on a fragment of a folded structure; each gray box denotes a term that involves four nucleotides.

a tree, we view the process as a map from a point $\mathsf{t} = \langle \mathsf{b}, t \rangle$ on a tree $\mathcal{T}$ (defined by branch b and the time t within it) into a random variable $\boldsymbol{X}^{(\mathsf{t})}$. Similarly, we generalize the definition of the Markov-consistent density set η to include functions on trees. We define continuity of functions on trees in the obvious manner.

4.2 Inference on Trees

The variational approximation on trees is thus similar to the one on intervals. Within each branch, we deal with the same update formulas as in linear time. We denote by $\mu^i_{x_i}(\mathsf{b}, t)$ and $\rho^i_{x_i}(\mathsf{b}, t)$ the messages computed on branch b at time t. The only changes occur at vertices, where we cannot use the Euler-Lagrange equations (Appendix A.2), therefore we must derive the propagation equations using a different method.

The following lemma shows and proves the update equations for the parameters $\mu^i(t)$ and $\rho^i(t)$ at the vertices:

Proposition 4.2.1: *Given a vertex T with an incoming branch b_1 and two outgoing branches $\mathsf{b}_2, \mathsf{b}_3$. The following are the correct updates for our parameters $\mu^i_{x_i}(t)$ and $\rho^i_{x_i}(t)$:*

$$\rho^i_{x_i}(\mathsf{b}_1, T) = \rho^i_{x_i}(\mathsf{b}_2, 0)\rho^i_{x_i}(\mathsf{b}_3, 0) \tag{4.1}$$

$$\mu^i_{x_i}(\mathsf{b}_k, 0) = \mu^i_{x_i}(\mathsf{b}_1, T) \qquad k = 2, 3. \tag{4.2}$$

Proof: We denote the time at the vertex $t_0 = (\mathsf{b}_1, T)$, the time just before as $t_1 = (\mathsf{b}_1, T - h)$ and the times just after it on each branch $t_2 = (\mathsf{b}_2, h)$ and $t_3 = (\mathsf{b}_3, h)$, as in Fig. 4.3.

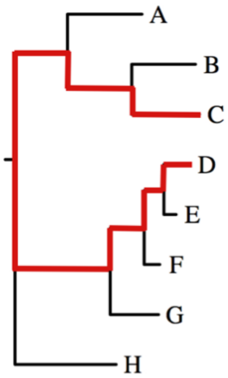

Figure 4.2: An example of a phylogenetic tree. Branch lengths denote time intervals between events with the interval used for the comparison in Fig. 4.5a highlighted.

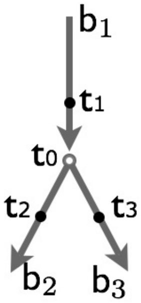

Figure 4.3: Branching process with discretization points of Lemma 4.2.1.

The marginals $\mu^i_{x_i}(\mathsf{b}_1, t)$ are continuous, as they are derived from the forward differential equation.

To derive the propagation formula for the $\rho^i_{x_i}(t)$ requires additional care. The $\rho^i_{x_i}(t)$ have been derived from the constraints on the time-derivative of $\mu^i_{x_i}(t)$. In a vertex this constraint is threefold, as we now have the constraints on b_1

$$\frac{\mu^i_{x_i}(t_0) - \mu^i_{x_i}(t_1)}{h} = \sum_{y_i} \gamma^i_{x_i,y_i}(t_1)$$

and those on the other branches b_k for $k = 2, 3$

$$\frac{\mu^i_{x_i}(t_k) - \mu^i_{x_i}(t_0)}{h} = \sum_{y_i} \gamma^i_{x_i,y_i}(t_0) \quad .$$

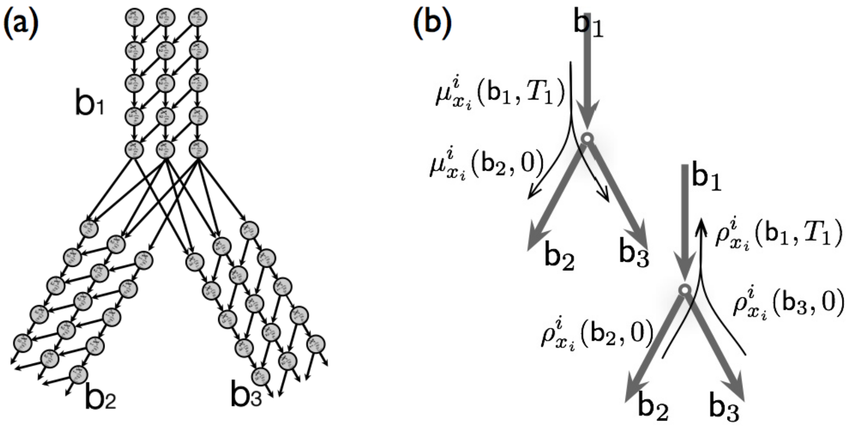
(a)
b1
b2
b3
(b)
b1
$\mu^i_{x_i}(\mathsf{b}_1, T_1)$
$\mu^i_{x_i}(\mathsf{b}_2, 0)$
b2
b3
b1
$\rho^i_{x_i}(\mathsf{b}_1, T_1)$
$\rho^i_{x_i}(\mathsf{b}_3, 0)$
$\rho^i_{x_i}(\mathsf{b}_2, 0)$
b2
b3

Figure 4.4: Structure of the branching process. **(a)** The discretized CTBN underlying the process in an intersection. **(b)** Illustration of the ODE updates on a directed tree, updating $\rho^i(t)$ backwards using (4.1) and $\mu^i(t)$ forwards using (4.2).

The integrand of the Lagrangian corresponding to point t_0 is

$$\begin{aligned}\mathcal{L}_{|t_0} &= \tilde{\mathcal{F}}(\eta;\mathbb{Q})_{|t_0} + \lambda^0(t_1)\left(\frac{\mu^i_{x_i}(t_0)-\mu^i_{x_i}(t_1)}{h} - \sum_{y_i}\gamma^i_{x_i,y_i}(t_1)\right)\\ &- \sum_{k=2,3}\lambda^k(t_0)\left(\frac{\mu^i_{x_i}(t_k)-\mu^i_{x_i}(t_0)}{h} - \sum_{y_i}\gamma^i_{x_i,y_i}(t_0)\right)\end{aligned}$$

and because this is the only integrand which involves $\mu_{x_i}(t_0)$ the derivative of the Lagrangian collapses into

$$\begin{aligned}\frac{\partial}{\partial\mu^i_{x_i}(t_0)}\mathcal{L} &= \frac{\partial}{\partial\mu^i_{x_i}(t_0)}\mathcal{L}_{|t_0}\\ &= \frac{\lambda^0(t_1)}{h} - \left(\frac{\lambda^2(t_0)}{h} + \frac{\lambda^3(t_0)}{h}\right) + \frac{\partial}{\partial\mu^i_{x_i}(t_0)}\tilde{\mathcal{F}}(\eta;\mathbb{Q})_{|t_0} = 0 \ .\end{aligned}$$

Rearranging the previous equation and multiplying by h, we get

$$\lambda^0(t_1) = \lambda^2(t_0) + \lambda^3(t_0) + \frac{\partial}{\partial\mu^i_{x_i}(t_0)}\tilde{\mathcal{F}}(\eta;\mathbb{Q})_{|t_0}h \ .$$

Looking at (3.10) we can see that as t_0 is not a leaf, and thus $\mu^i_{x_i}(t_0) > 0$ and the derivative of the functional cannot diverge. Therefore as $h \to 0$ this term vanishes and we are left with

$$\lambda^0(t_1) = \lambda^2(t_0) + \lambda^3(t_0)$$

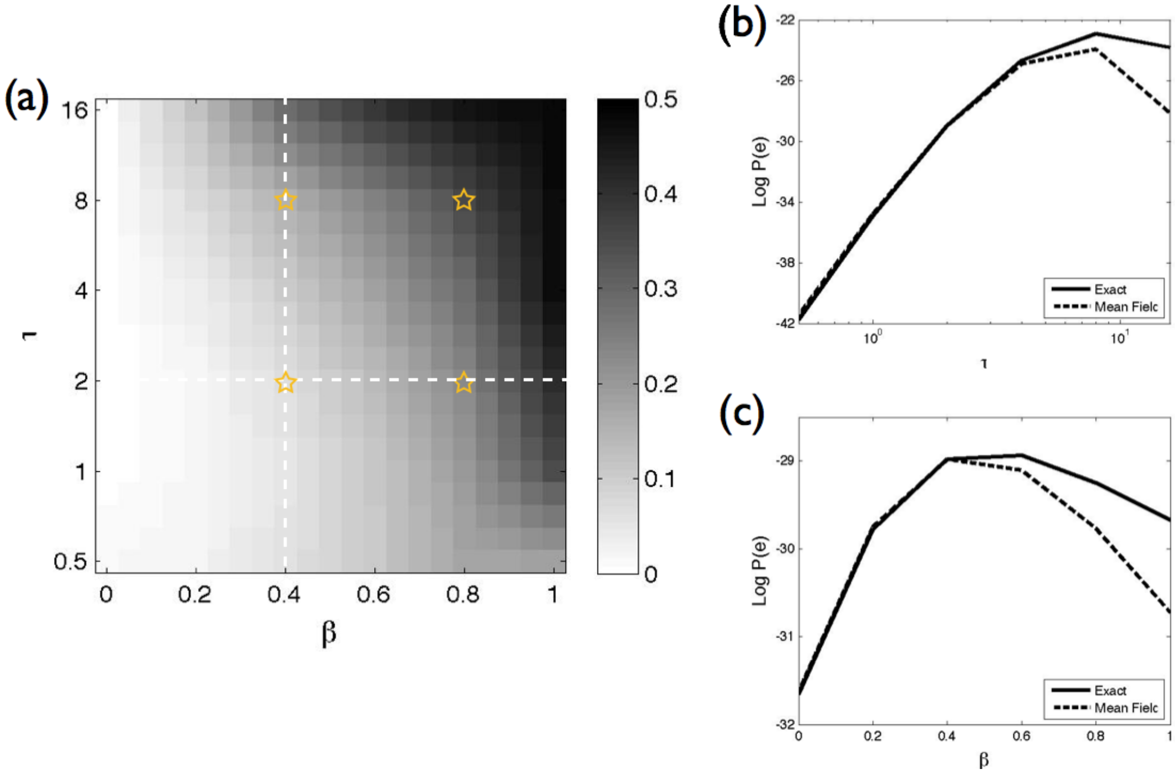

Figure 4.5: **(a)** Evaluation of the relative error in expected sufficient statistics for an Ising chain in branching-time; compare to Fig. 3.5(a). **(b)**,**(c)** Evaluation of the estimated likelihood on a tree w.r.t. the rate τ and coupling β; compare to Fig. 3.5(b),(c).

which after taking exponents gives us (4.1). ∎

Using Proposition 4.2.1 we can set the updates of the different parameters in the branching process according to (4.1–4.2). In the backward propagation of ρ^i, the value at the end of b_1 is the product of the values at the start of the two outgoing branches. This is the natural operation when we recall the interpretation of ρ^i as the probability of the downstream evidence given the current state (which is its exact meaning in a single component process): the downstream evidence of b_2 is independent of the downstream evidence of b_3, given the state of the process at the vertex T. The forward propagation of μ^i simply uses the value at the end of the incoming branch as initial value for the outgoing branches.

When switching to trees, we essentially increase the amount of evidence about intermediate states. Consider for example the tree of Fig. 4.2 with an Ising chain model (as in the previous section). We can view the span from C to D as an interval with evidence at its end. When we add evidence at the tip of other branches we gain more information about intermediate points between C and D. Even though this evidence can represent evolution from these intermediate points, they do change our information state about them. To evaluate the impact of these changes on our

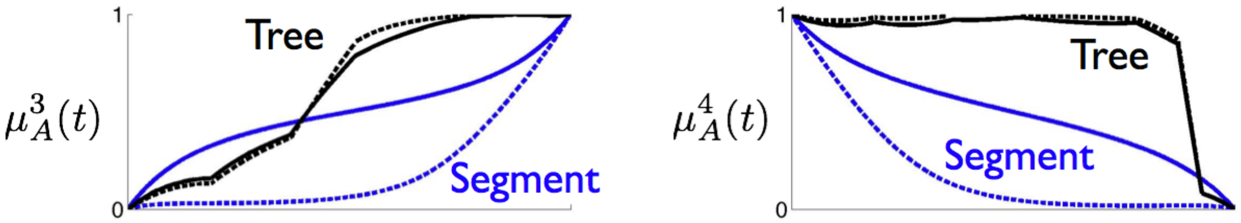

Figure 4.6: Comparison of exact vs. approximate inference along the branch from C to D in the tree of Fig. 4.2 with and without additional evidence at other leaves. Exact marginals are shown in solid lines, whereas approximate marginal are in dashed lines. The two panels show two different components.

approximation, we considered the tree of Fig. 4.2, and compared it to inference in the backbone between C and D (Fig. 3.5). Comparing the true marginal to the approximate one along the main backbone (see Fig. 4.6) we see a major difference in the quality of the approximation. The evidence in the tree leads to a much tighter approximation of the marginal distribution. A more systematic comparison (Fig. 4.5a,b) demonstrates that the additional evidence reduces the magnitude of the error throughout the parameter space.

As a more demanding test, we applied our inference procedure to the model introduced by Yu and Thorne (2006) for a stem of 18 interacting RNA nucleotides in 8 species in the phylogeny of Fig. 4.2. We compared our estimate of the expected sufficient statistics of this model to these obtained by the Gibbs sampling procedure of El-Hay et al. (2008). The results, shown in Fig. 4.7, demonstrate that over all, the two approximate inference procedures are in good agreement about the value of the expected sufficient statistics.

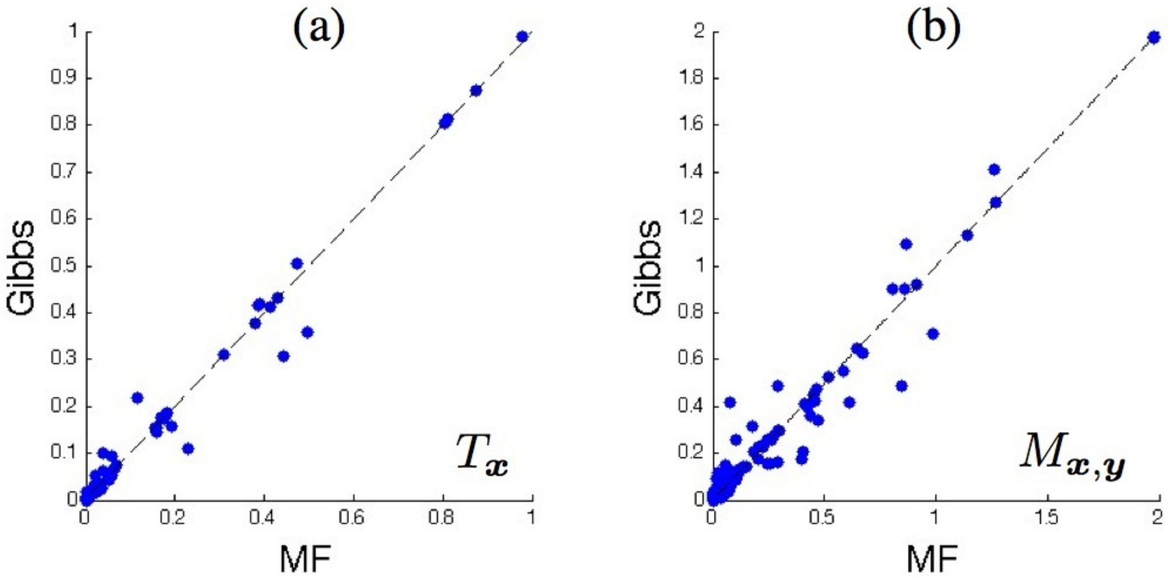

Figure 4.7: Comparison of estimates of expected sufficient statistics in the evolution of 18 interacting nucleotides, using a realistic model of RNA evolution. Each point is an expected statistic value; the x-axis is the estimate by the variational procedure, whereas the y-axis is the estimate by Gibbs sampling.

Chapter 5

Discussion

In this dissertation we formulate a general variational principle for continuous-time Markov processes and use it to derive an efficient procedure for inference in CTBNs. In this mean field-type approximation, we use a product of independent inhomogeneous processes to approximate the multi-component posterior.

5.1 Related Works

Variational approximations for different types of continuous-time processes have been recently proposed (Opper and Sanguinetti, 2007; Archambeau et al., 2008). Our approach is motivated by results of Opper and Sanguinetti (2007) who developed a variational principle for a related model. Their model, which they call a Markov jump process, is similar to an HMM, in which the hidden chain is a continuous-time Markov process and there are (noisy) observations at discrete points along the process. They describe a variational principle and discuss the form of the functional when the approximation is a product of independent processes. There are two main differences between the setting of Opper and Sanguinetti and ours. First, we show how to exploit the structure of the target CTBN to reduce the complexity of the approximation. These simplifications imply that the update of the i'th process depends only on its Markov blanket in the CTBN, allowing us to develop efficient approximations for large models. Second, and more importantly, the structure of the evidence in our setting is quite different, as we assume deterministic evidence at the end of intervals. This setting typically leads to a posterior Markov process (recall that the posterior is still Markovian), in which the instantaneous rates used by Opper and Sanguinetti diverge toward the end point—the rates of transition into the observed state go to infinity, while transitions into states inconsistent with the evidence decay to zero. Opper and Sanguinetti use instantaneous rates as the variational parameters, which in our framework lead to numerical problems at the end points. We circumvent this problem by using the marginal density representation which is much more stable numerically.

In comparison to the Expectation Propagation procedure of Nodelman et al. (2005b) our algorithm posseses several important advantages. The Mean Field procedure is guarantied to converge to a consistent distribution. Due to this consistency, the calculated energy functional is a lower bound on the likelihood of the evidence. Additionally, our algorithm is fully adapted to the continuous-time representation and unlike EP does not require fine-tuning of any parameters. Even though we do not directly model dependencies as in the EP algorithm, we are able to give a good approximation of the expected sufficient statistics in trees.

5.2 Extensions

A possible extenstion is to use our variational procedure to generate the initial distribution for the Gibbs sampling procedure and thus skip the initial burn-in phase and produce accurate samples.

Another attractive aspect of this new variational approximation is its potential use for learning model parameters from data. It can be easily combined with the EM procedure for CTBNs (Nodelman et al., 2005a), to obtain a Variational-EM procedure for CTBNs, which monotonically increases the likelihood by alternating between steps that improve the approximation η (the updates discussed here) and steps that improve the model parameters θ (an M-step (Nodelman et al., 2005a)).

5.3 Conclusions

Our procedure enjoys the same benefits encountered in discrete time mean field procedure (Jordan et al., 1998): it provides a lower-bound on the likelihood of the evidence and its run time scales linearly with the number of components. Using asynchronous updates it is guaranteed to converge, and the approximation represents a consistent joint distribution. It also suffers from expected shortcomings: the functional has multiple local maxima, it cannot capture complex interactions in the posterior (Example 2.3.1). By using a time-inhomogeneous representation, our approximation does capture complex patterns in the temporal progression of the marginal distribution of each component. Importantly, the continuous time parametrization enables straightforward implementation using standard ODE integration packages that automatically tune the trade-off between time granularity and approximation quality. We show how to extend it to perform inference on phylogenetic trees, and show that it provides fairly accurate answers in the context of a real application (Fig. 4.7).

One of the key developments here is the shift from (piecewise) homogeneous parametric representations to continuously inhomogeneous representations based on marginal density sets. This shift increases the flexibility of the approximation and, somewhat surprisingly, also significantly simplifies the resulting formulation.

Appendix A

Supplementary Details

A.1 Constrained Optimization

Given a multidimensional function $f(\boldsymbol{x})$ over $\boldsymbol{x} \in \mathcal{X}$ we would like to find

find $\quad \max_{\boldsymbol{x}} f(\boldsymbol{x})$
subject to $\quad h_i(\boldsymbol{x}) = 0$ for $i = 1, \ldots, K$.

To maximize subject to the K equality constraints, we introduce new auxilliary variables named *Lagrange multipliers*. We define the *Lagrangian*

$$\mathcal{L}(\boldsymbol{x}, \lambda_i) = f(\boldsymbol{x}) - \sum_{i=1}^{K} \lambda_i h_i(\boldsymbol{x})$$

which allows us to incorporates the constraints, as well as the function itself. Now the original constrained problem turns into the following optimization problem

find $\quad \min_{\lambda_i, \boldsymbol{x}} \quad \mathcal{L}(\boldsymbol{x}, \lambda_i)$.

The partial derivatives of the lagrangian w.r.t. its parameters are

$$\begin{aligned} \frac{\partial}{\partial \boldsymbol{x}} \mathcal{L}(\boldsymbol{x}, \lambda_i) &= \frac{\partial}{\partial \boldsymbol{x}} f(\boldsymbol{x}) - \sum_{i=1}^{K} \lambda_i \frac{\partial}{\partial \boldsymbol{x}} h_i(\boldsymbol{x}) \\ \frac{\partial}{\partial \lambda_i} \mathcal{L}(\boldsymbol{x}, \lambda_i) &= -h_i(\boldsymbol{x}) \ . \end{aligned}$$

We denote $\mathcal{U} \in \mathcal{X}$ to be the subspace of $\mathcal{X}$ that for each $\boldsymbol{x} \in \mathcal{U}$, $\boldsymbol{x}$ satisfies the constraints over all $h_i(\boldsymbol{x})$. The main strength of this formulation comes from the following lemma (Bertsekas, 1982)

Lemma A.1.1: *An interior point $\boldsymbol{x} \in \mathcal{U}$ is a stationary point of the function $f(\boldsymbol{x})$ iff all the partial derivatives of $\mathcal{L}(\boldsymbol{x}, \lambda_i)$ vanish.*

This lemma gives us a new optimization problem - finding the fixed points of the Lagrangian will give us the constrained fixed points of the function.

This theory can also be generalized to functionals (as in our case), and inequality constraints.

A.2 Euler-Lagrange equations

The problem of finding the fixed points of *functionals* comes from the field of *Calculus of variations*, as opposed to ordinary calculus which deals with *functions*. As defined in Def. 2.1.4, a functional is a mapping from a linear space $\mathcal{X}$ to its underlying field. In our case the functional is our Lagrangian, which is an integral over real-valued functions, and the underlying field is $\mathbb{R}$. Given a functional of the form

$$I[y] = \int_a^b f(t, y(t), y'(t))dt$$

where $y'(t)$ is the time-derivative of the function $y(t)$. We would like to find a function $y(t)$ that minimizes (or in our case maximizes) the functional. For $y(t)$ to be a stationary point, it must satisfy the **Euler-Lagrange** equations (Gelfand and Fomin, 1963)

$$\frac{\partial}{\partial y} f(t, y(t), y'(t)) - \frac{d}{dt}\left(\frac{\partial}{\partial y'} f(t, y(t), y'(t))\right) = 0 \ . \tag{A.1}$$

An example for the use of this equation is in (3.10).

A.3 Numerical Solutions of Differential Equations

Numerical integration is the calculation of an integral using numerical techniques, and in this section we will talk about solving *definite integrals*. In our case, we are given a *differential equation* which is defined by an initial value, and the derivative of the function. Our goal is to calculate the values of $f(t)$ for each $t \in [0, T]$. For a given positive h, we can write the discretized differential equation as derived from the first *taylor expansion* of $f(t)$:

$$f(t + h) = f(t) + \frac{d}{dt} f(t) \cdot h \ . \tag{A.2}$$

To calculate the values of $f(t)$ at each $t \in [0, T]$, we have to perform sequential integration forwards from $t = 0$ to $t = T$.

For example, the *forward master equation* of the marginals $\mu^i_{x_i}(t)$ is given by the initial value $\mu^i_{x_i}(0) = \delta_{x_i,e_{i0}}$ and the derivative

$$\frac{d}{dt}\mu^i_{x_i}(t) = \mu_{x_i}(t)\sum_{y_i} \bar{q}^i_{x_i,y_i}(t)\frac{\rho^i_{y_i}(t)}{\rho^i_{x_i}(t)} \ .$$

which leads us to the first order expansion

$$\mu^i_{x_i}(t + h) = \mu_{x_i}(t)\left(1 + \sum_{y_i} \bar{q}^i_{x_i,y_i}(t)\frac{\rho^i_{y_i}(t)}{\rho^i_{x_i}(t)} \cdot h\right) \ .$$

We see that to calculate $\mu^i_{x_i}(t+h)$ we must have all the values of $\mu^i_{y_i}(t)$ for each y_i, as well as the values of $\bar{q}^i(t)$ and $\rho^i(t)$ which have already been calculated in the backward pass of $\rho^i(t)$. Starting from $t = 0$, we set $\mu^i(0)$ to be consistent with the evidence. Then we propagate the values of $\mu^i(t+h)$ for every t in the interval.

In this section we will address two parts of the numerical integration procedure. The first deals with how to choose the points in which we calculate the values of the function, and the second deals with how we calculate the value of the function in a given point.

Adaptive Point Selection The formulation of the numerical solution of an integral with (A.2) leads us to the simplest numerical methods which use the *Newton-Cotes* integration formulas. In these methods we calculate the values of a function $f(t)$ in constantly spaced points $0, h, \ldots, T-h, T$, similar to discretizing the function along regular intervals. In general, these methods are useful for the case where we can only determine the derivative of $f(t)$ at these certain intervals, and not in between.

Due to the continuous nature of $f(t)$, we expect an error of a certain magnitude, given that the derivative need not be constant in between the calculated points. However, the error generated from using this construction is guarantied to go to 0 as $h \to 0$, but at a higher and higher computational cost. These methods do not utilize prior knowledge of the function's dynamics, nor its smoothness. For instance, the calculation of a linear function in an interval $[0, T]$ where $f(0)$ is known and $\frac{d}{dt}f(t) = C$ will be in $\Theta(\frac{T}{h})$ in this naive method (expensive, even though with no error). On the other hand, understanding that the function is linear brings us to the obvious solution that $f(t) = f(0) + t \cdot \frac{d}{dt}f(0)$, which can be calculated in $\Theta(1)$ without error. This insight motivates us to move away from the discretized methods and back into the continuous framework.

In contrast to the constant interval methods, we would like our integrator to determine the intervals in which to calculate the function's value adaptively. A function whose derivative changes slowly will require a small number of sample points, while a highly dynamic one will need many closely-positioned points. The adaptive integrator is able to bound the error of the calculation by choosing the appropriate step size at each t. There are a multitude of ways to achieve these requirements, the inquisitive reader is referred to (Forsythe et al., 1977). The saving in computation can be substantial, as we see in Fig. A.1 where a continuous function is computed using numerical integration by discretization (blue) and adaptively (red).

In our work we were able to utilize the continuous nature of the approximation and thus use adaptive numerical integration. We show results (Fig. 3.7) that using this adaptive integration procedure is superior to any constant interval discrete approximation, and achieves the best trade-off between computation and error.

Function Value Calculation The second issue has to do with the calculation of $f(t)$ for each t in the interval. The most naive method to approximate $f(t+h)$ given $f(t)$ is the rectangle rule (also named "Euler's method"), which simply uses (A.2). Not surprisingly, this approximation gives a cubic error of $O(h^2)$ in the interval length of h. More complex variants of this rule are known, such as the "trapezoid" rule, or the different "quadratures" which approximate the next value of $f(t+h)$ using splines or other more complex approximations instead of linear extrapolation.

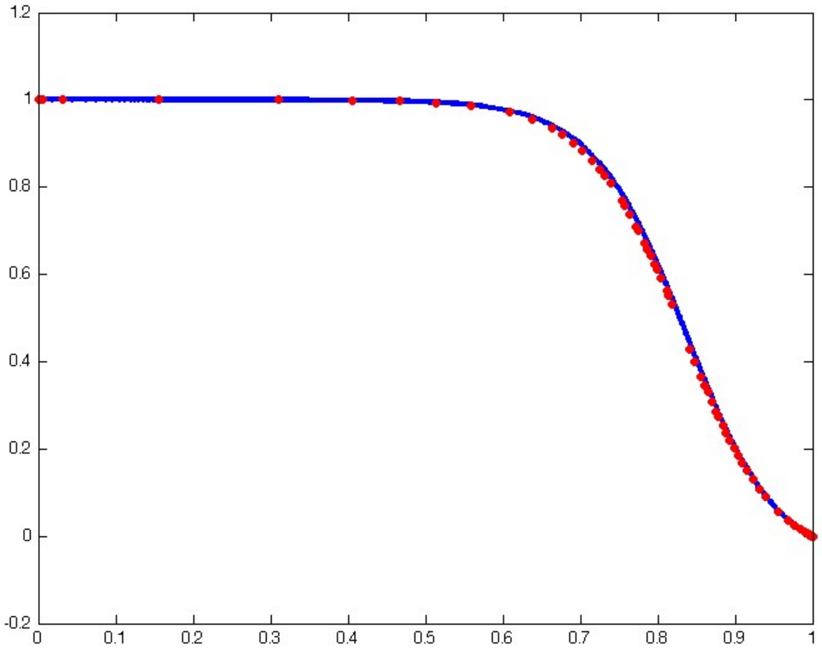

Figure A.1: Comparison of numerical integration between Newton-Cotes (blue) and adaptive (red). The number of integration points varies according to the second derivative of the calculated function, which near 0 is approximately linear and near 1 has an exponential behaviour. The number of adaptive integration points is smaller than the discretization method's by an order of magnitude, with a miniscule additional error.

In our work we used the *Runge-Kutta-Fehlberg* (RKF45) algorithm (Num, 2007) in its adaptive version, implemented by the GNU Scientific Library (www.gnu.org/software/gsl/). This method achieves a $O(h^5)$ error bound by factoring in the derivatives of several neighboring points for a much more accurate approximation.

Bibliography

(2007, September). *Numerical Recipes 3rd Edition: The Art of Scientific Computing* (3 ed.). Cambridge University Press.

Archambeau, C., M. Opper, Y. Shen, D. Cornford, and J. Shawe-Taylor (2008). Variational inference for diffusion processes. In J. Platt, D. Koller, Y. Singer, and S. Roweis (Eds.), *Advances in Neural Information Processing Systems 20*, pp. 17–24. Cambridge, MA: MIT Press.

Bertsekas, D. P. (1982). *Constrained Optimization and Lagrange Multiplier Methods*. London, New York: Academic Press.

Boyen, X. and D. Koller (1998). Tractable inference for complex stochastic processes. In *Proceedings of the Fourteenth Conference on Uncertainty in AI (UAI)*, pp. 33–42.

Chung, K. (1960). *Markov chains with stationary transition probabilities*. Berlin: Springer Verlag.

Dean, T. and K. Kanazawa (1989). A model for reasoning about persistence and causation. Technical report, Providence, RI, USA.

El-Hay, T., N. Friedman, D. Koller, and R. Kupferman (2006). Continuous time markov networks. In *Proceedings of the Twenty-second Conference on Uncertainty in AI (UAI)*.

El-Hay, T., N. Friedman, and R. Kupferman (2008). Gibbs sampling in factorized continuous-time markov processes. In *Proceedings of the Twenty-Fourth Conference on Uncertainty in AI (UAI)*.

Elidan, G., I. Mcgraw, and D. Koller (2006). Residual belief propagation: informed scheduling for asynchronous message passing. In *Uncertainty in Artificial Intellignece*.

Fan, Y. and C. Shelton (2008). Sampling for approximate inference in continuous time Bayesian networks. In *Tenth International Symposium on Artificial Intelligence and Mathematics*.

Fan, Y. and C. R. Shelton (2009). Learning continuous-time social network dynamics. In *Proceedings of the Twenty-Fifth International Conference on Uncertainty in Artificial Intelligence*.

Feller, W. (1968). *An introduction to Probability Theory and its Applications*, Volume 1.

Felsenstein, J. (2004). *Inferring Phylogenies*. Sinauer.

Forsythe, G. E. G. E., M. A. Malcolm, and C. B. Moler (1977, May). *Computer Methods for Mathematical Computations*. Prentice-Hall series in automatic computation.

Fromer, M. and C. Yanover (2009). Accurate prediction for atomic-level protein design and its application in diversifying the near-optimal sequence space. *Proteins: Structure, Function, and Bioinformatics 75*, 682–705.

Gardiner, C. (2004). *Handbook of stochastic methods* (third ed.). New-York: Springer-Verlag.

Gelfand, I. M. and S. V. Fomin (1963). *Calculus of variations*. Revised English edition translated and edited by Richard A. Silverman.

Jordan, M. I., Z. Ghahramani, T. Jaakkola, and L. K. Saul (1998). An introduction to variational approximations methods for graphical models. In *Learning in Graphical Models*.

Koller, D. and N. Friedman (2009). *Probabilistic Graphical Models: Principles and Techniques*. Cambridge, MA: MIT Press.

Moler, C. B. and C. F. Van Loan (1978, October). Nineteen dubious ways to compute the exponential of a matrix. *20*(4), 801–836.

Murphy, K. P., Y. Weiss, and M. I. Jordan (1999). Loopy belief propagation for approximate inference: An empirical study. In *In Proceedings of Uncertainty in AI*, pp. 467–475.

Nodelman, U., C. Shelton, and D. Koller (2002). Continuous time Bayesian networks. In *Proc. Eighteenth Conference on Uncertainty in Artificial Intelligence (UAI '02)*, pp. 378–387.

Nodelman, U., C. Shelton, and D. Koller (2003). Learning continuous time Bayesian networks. In *Proc. Nineteenth Conference on Uncertainty in Artificial Intelligence (UAI '03)*, pp. 451–458.

Nodelman, U., C. Shelton, and D. Koller (2005a). Expectation maximization and complex duration distributions for continuous time Bayesian networks. In *Proc. Twenty-first Conference on Uncertainty in Artificial Intelligence (UAI '05)*, pp. 421–430.

Nodelman, U., C. Shelton, and D. Koller (2005b). Expectation propagation for continuous time Bayesian networks. In *Proc. Twenty-first Conference on Uncertainty in Artificial Intelligence (UAI '05)*, pp. 431–440.

Opper, M. and G. Sanguinetti (2007). Variational inference for Markov jump processes. In *NIPS*.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann.

Saria, S., U. Nodelman, and D. Koller (2007). Reasoning at the right time granularity. In *Proceedings of the Twenty-Third Conference on Uncertainty in AI (UAI)*.

Talya Meltzer, A. G. and Y. Weiss (2009). Convergent message passing algorithms - a unifying view. In *Proc. Twenty-eighth Conference on Uncertainty in Artificial Intelligence (UAI '09)*.

Wainwright, M. J. and M. Jordan (2008). Graphical models, exponential families, and variational inference. *Found. Trends Mach. Learn. 1*, 1–305.

Yedidia, J. S., W. T. Freeman, and Y. Weiss (2005). Constructing free-energy approximations and generalized belief propagation algorithms. *Information Theory, IEEE Transactions on 51*(7), 2282–2312.

Yu, J. and J. L. Thorne (2006). Dependence among sites in RNA evolution. *Mol. Biol. Evol. 23*, 1525–37.