Mean field analysis for Continuous Time Bayesian Networks

Davide Cerotti*, Daniele Codetta-Raiteri

DiSIT, Computer Science Institute, Università Piemonte Orientale, Alessandria, Italy
{davide.cerotti, daniele.codetta}@uniupo.it

Abstract. In this paper we investigate the use of the mean field technique to analyze Continuous Time Bayesian Networks (*CTBN*). They model continuous time evolving variables with exponentially distributed transitions with the values of the rates dependent on the parent variables in the graph. *CTBN* inference consists of computing the probability distribution of a subset of variables, conditioned by the observation of other variables' values (evidence). The computation of exact results is often unfeasible due to the complexity of the model. For such reason, the possibility to perform the *CTBN* inference through the equivalent Generalized Stochastic Petri Net (*GSPN*) was investigated in the past. In this paper instead, we explore the use of mean field approximation and apply it to a well-known epidemic case study. The *CTBN* model is converted in both a *GSPN* and in a mean field based model. The example is then analyzed with both solutions, in order to evaluate the accuracy of the mean field approximation for the computation of the posterior probability of the *CTBN* given an evidence. A summary of the lessons learned during this preliminary attempt concludes the paper.

Acronym list:

BN	Bayesian Network
CIM	Conditional Intensity Matrix
CTBN	Continuous Time Bayesian Network
CTMC	Continuous Time Markov Chain
DAG	Directed Acyclic Graph
DBN	Dynamic Bayesian Network
GSPN	Generalized Stochastic Petri Net
MFM	Mean Field Model
SIR	Susceptible-infected-recovered model

1 Introduction

Temporal probabilistic graphical models allow for a factorization of the state space of a process, resulting in improved modeling and inference features. Usually such models are based on graph structures and analyzed according to the

* corresponding author

principles of the Bayesian theory. When time is considered continuous, *Continuous Time Bayesian Networks* (*CTBN*) [16] allow us to represent variables whose behavior depends on their parent variables.

Actually, exact inference in large *CTBN* may often be impractical, so approximations through message-passing algorithms on cluster graphs [20, 15], or through sampling [10, 11] have been proposed. The generation of large models arises in several applications, such as biological process modeling or reliability analysis of distributed systems where a numerous collection of similar entities or subcomponents influence their behavior by interacting with each other.

In [9] the authors propose to perform inference on a *CTBN* by describing it in terms of a *Generalized Stochastic Petri Nets* (*GSPN*) [1] according to a set of translation rules and inference algorithms. In this paper instead, we investigate an alternative type of approximation grounded on the mean field theory [6, 18] which allows us to compute the exact behavior of this kind of models when the number of entities tends to infinity, and provides an approximation in case the number of entities is large enough. Exploiting the *decoupling assumption* [8], i.e. that many-to-many stochastic interacting systems can be analyzed as one tagged stochastic entity interacting with a single deterministic one, allows us to solve each submodel in isolation and avoid the construction of a large state space with a significant reduction in the computational effort of the analysis.

The main goals of the paper are: 1) exploring mean field application to *CTBN* models where the conditional dependencies between variables are defined over a reduced set of neighbor ones; 2) studying the impact of such approximation to the results of the inference computations.

To this end, Section 2 introduces the main concepts of *CTBN* and mean field theory used in the paper. Section 3 describes a well-known case study of epidemic propagation and presents the resulting models expressed in terms of a *CTBN*, its conversion into *GSPN* and its mean-field approximation. Section 4 briefly illustrates the two main inference tasks in *CTBN*, namely *prediction* and *smoothing*, focusing on the computation of the former through evaluation of the *GSPN* and the *mean field model* (*MFM*). Section 5 presents the evaluation of the accuracy of the results of the inference provided by the mean field approximation. Finally, Section 6 concludes the paper with a discussion of the effectiveness of the approach and proposes further directions of the work.

2 Preliminary notions

2.1 Continuous Time Bayesian Networks

Probabilistic graphical models for reasoning about processes that evolve over time, permit a *factorization* [13] of the state space of the process, resulting in better modeling and inference features. Such models are usually based on graph structures, grounded on the theory of *Bayesian Networks* (*BN*). When time is taken into account, the main choice concerns whether to consider it as a discrete or a continuous dimension. In the first case, models like *Dynamic*

Bayesian Networks (*DBN*) [14] have become a natural choice; however, there is not always an obvious discrete time unit and, when the process is characterized by several components evolving at different rates, the finer granularity dictates the rules for the discretization [19]. Moreover, if evidence is irregularly spaced in time, all the intervening time slices still have to be dealt with (even if no evidence is available at a given time point). For these reasons, models based on *BN*, but with a continuous time representation of the temporal evolution have started to be investigated. *Continuous Time Bayesian Networks* (*CTBN*) have been firstly proposed in [16, 17] and then refined in [20]. Extensions have also been proposed both regarding the use of indirect graph models [10] and the use of Erlang-Coxian distributions on the transition time [12].

Following the original paper [16], a *CTBN* is defined as follows: let $V = \{X_1, \ldots, X_n\}$ be a set of discrete variables, a *CTBN* over V consists of two components. The first one is an initial distribution P_V^0 over V. The second component is a continuous-time transition model specified as:

- a directed graph G whose nodes are $X_1, \ldots, X_n$ ($Pa(X_i)$ denotes the parents of X_i in G);
- a *Conditional Intensity Matrix* (*CIM*) $Q_{X_i|Pa(X_i)}$ for every $X_i \in V$. The *CIM* of a variable X_i provides the transition rates for each possible pair of values of X_i, given any possible combination of the parent nodes' values.

In other words, each node (variable) X_i incorporates a *Continuous Time Markov Chain* (*CTMC*) having as many states as the possible values of X_i; in the *CTMC*, the state transition rates depend on the current values of $Pa(X_i)$, i.e. the parent nodes of X_i. With respect to standard *BN* having an acyclic graph structure (*DAG*), cycles are instead permitted in *CTBN* where a node (variable) X_i, ancestor of X_j, can be reachable from X_j. A cycle could be even composed by one node X_i: $X_i \in Pa(X_i)$.

2.2 An illustrative example

We now consider a simple case study which can be easily modelled in form of *CTBN*. It consists of a small system composed by the main component A and its "warm" spare component B. This means that initially both components are working, but A is active while B is dormant; in case of failure of A, B is activated in order to replace A in its function. The expression "warm" referred to the spare component B indicates that the probability of failure of B is not null while B is dormant, and such value is increased while B is active. So, B may fail as well, and this may happen before or after the failure of A; if B fails before A, B cannot replace A.

The time to failure of a component is a random variable ruled by the negative exponential distribution whose parameter is called failure rate and is the inverse of the mean time to failure. In the case of the main component A, the failure rate is λ_A =1.0E-06 h^{-1}. The failure rate of B, λ_B, changes according to its current state: if B is dormant (A is not failed) λ_B is equal to 5.0E-07 h^{-1}; if

instead B is active (A is failed), λ_B is equal to 1.0E-06 h^{-1}. In other words, the failure rate of B is reduced with respect to the failure of A, while B is dormant; if instead B is active in order to replace A, the failure rate of B is the same as the failure rate of A.

We make two assumptions about repair: 1) only while B is active (to replace A), A undergoes repair; 2) only while A is working, B undergoes repair (if B is failed). The time to repair a component is a random variable, still ruled by the negative exponential distribution having as parameter the repair rate equal to the inverse of the mean time to repair the component. A and B have the same repair rate $\mu_A = \mu_B = 0.02\ h^{-1}$ if the repair is enabled (as explained above); if not, $\mu_A = \mu_B = 0$.

The *CTBN* model. The case study described above is represented by the *CTBN* model in Figure 1 where the variables A and B represent the states of the corresponding components. Both variables are binary because each entity can be in the working state or in the failed state (for the component B, the working state comprises both the dormancy and the activation). In particular, we represent the working state with the value 1, and the failed state with the value 2.

The arc from A to B establishes that the variable A influences the variable B because the failure rate of the component B depends on the state of A, as described above. Moreover, the possibility of repairing B depends on the current state of A. The arc from B to A expresses that B influences A because the possibility of repairing A depends on the current state of B.

Both variables implicitly incorporate a *CTMC* composed by two states: 1 (working) and 2 (failed). The initial probability distribution holds for both variables, and establishes that both components are initially supposed to be working. The *CTMC* is shown in Figure 2: the transition from 1 to 2 occurs after a random period of time ruled by the negative exponential distribution according to the failure rate λ. The transition from 2 to 1 is ruled by the same distribution, but according to the repair rate μ.

In the case of A, the current value of the rate μ_A depends on the current value of the variable B influencing A. This is shown by the *CIM* reported in Table 1.a where we can notice that the rate μ_A is not null only if the state of B is 1. The rate λ_A instead, is constant (the state B influences only the repair of the component A). In the case of the variable B, both the rates λ_B and μ_B depend on the current value of the variable A, as shown by the *CIM* appearing in Table 1.b where λ_B is increased when A is equal to 2. As in the case of A, the rate μ_B is not null only if the value of A is 1.

2.3 Mean field theory

Let us consider a generic model of $N \in \mathbb{N}$ identical interacting objects, where each of them has a state and interacts with others according to the Markovian property, i.e. the evolution of the system depends only on the collection of

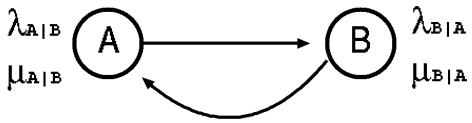

Fig. 1. The *CTBN* model of the case study.

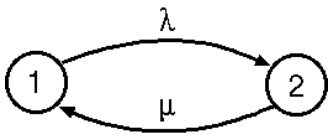

Fig. 2. *CTMC* incorporated in the variables A and B in the *CTBN* model in Figure 1.

a)

$1 \to 2$		$2 \to 1$	
B	λ_A	B	μ_A
1	1.0E-06 h^{-1}	1	0.01 h^{-1}
2	1.0E-06 h^{-1}	2	0

b)

$1 \to 2$		$2 \to 1$	
A	λ_B	A	μ_B
1	5.0E-07 h^{-1}	1	0.01 h^{-1}
2	1.0E-06 h^{-1}	2	0

Table 1. a) *CIM* of A. b) *CIM* of B.

states at the current instant of time. The object with index $n \in \{1, 2, \ldots, N\}$ is represented by the stochastic process $\{X_n^N(t)\}$ which takes values in the set $\mathbb{S} = \{0, \ldots, K-1\}$ with $K = |\mathbb{S}|$ the number of different states. The complete system can be described by a multinomial stochastic process:

$$Y^N(t) = (X_1^N(t), \ldots, X_N^N(t)),$$

with a state space of K^N elements. Assuming that the objects are indistinguishable, it is sufficient to keep track of the proportion of objects in each state. These values define a related stochastic process $M^N(t) = (M_0(t), \ldots, M_{K-1}(t))$ called the *occupancy measure* whose elements are defined as:

$$M_i^N(t) = \frac{1}{N} \sum_{n=1}^{N} 1_{\{X_n^N(t)=i\}}, i \in \mathbb{S}$$

where the indicator function $1_{\{X_n^N(t)=i\}}$ is 1 if $X_n^N(t) = i$, 0 otherwise. A state of such a process is denoted by $\mathbf{m} = (m_0, \ldots, m_{K-1})$ where m_i is the fraction of objects in state i.

Under very general assumptions [2], the mean field convergence result states that when the number of objects N goes to infinity the occupancy measure converges to a deterministic limit $\mathbf{u}(t)$ (the so-called mean field), thus for each local state i the fraction of objects in state i at time t is known with probability one. Moreover, Sznitman proves the *Mean Field Independence*, or *Propagation of Chaos* theorem [21] that allows one to perform the following approximation:

$$Pr(X_1^N(t) = i_1, \ldots, X_N^N(t) = i_n) \approx u_{i_1}\left(\frac{t}{N}\right) \cdot \ldots \cdot u_{i_n}\left(\frac{t}{N}\right) \tag{1}$$

where u_{i_j} are the components of the deterministic limit $\mathbf{u}(t)$. Equation 1 exploits the *decoupling assumption* to approximate the behavior of the multinomial stochastic process as a product of the components of the deterministic limit.

3 A motivating case study

In this section we present a well-known *susceptible-infected-recovered* (*SIR*) model which will be used as a running example for showing how to: 1) describe it by a *CTBN* to be converted into *GSPN*; 2) apply the mean field approximation.

We consider a system of N nodes, each node may be in three possible states: susceptible (S), infected (I) or recovered (R). The nodes are vertices of a $2D$ torus graph, as shown in Figure 3(b), so that each of them is connected with their nearest neighbors and, when belonging to edges, with their corresponding nodes on the opposite edges of the grid. We denote by $X_i(\mathbf{v}, t) \in [0, 1]$ the probability that node in position $\mathbf{v}$ is in state $i \in \{S, I, R\}$ at time t. By definition, for each $\mathbf{v}$ and t, we have that $X_S(\mathbf{v}, t) + X_I(\mathbf{v}, t) + X_R(\mathbf{v}, t) = 1$.

The dynamic of the system is ruled by a set of exponential transitions as following. A susceptible node can become infected from an external source, with rate α, or from its neighbors, with a rate ρ times the number of current infected nearest neighbors $\eta(\mathbf{v}, t)$. An infected node becomes recovered at rate β and a recovered one turns to susceptible at rate γ. The resulting *CTMC* is depicted in Figure 3.a.

3.1 The *CTBN* model

The description of the *SIR* model as a *CTBN* (Figure 3.b) is straightforward: the current states of the nodes in position $\mathbf{v}_j$ are mapped to variables V_j with values 1, 2, 3, corresponding to the states S, I, R, respectively. Each variable (node) of the *CTBN* model incorporates the *CTMC* shown in Figure 3.a, where the state transitions having a null rate are not depicted. The initial probability distribution of each variable is shown in Table 2.a. The stochastic behavior of each node is conditioned by the values of the variables representing its nearest neighbors. In particular, the current rate of the transition from state 1 (S) to 2 (I) is dependent on the number of current infected nearest neighbors. We can define the *CIM* according to the value $\eta(\mathbf{v}, t)$, as shown in Table 2.b. The other state transitions (from I to R, and from R to S) have a constant (independent) rate (Table 2.b).

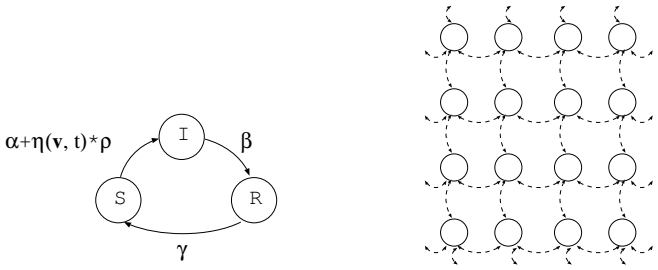

Fig. 3. a) The incorporated *CTMC*. b) The *CTBN* for the *SIR* model with bidirectional dependencies in a $2D$ torus graph.

V_j	Prob
S	1
I	0
R	0

(a)

$S \to I$		$I \to R$	$R \to S$
$\eta(\mathbf{v}, t)$	Rate		
k	$(\alpha + k\rho)$	β	γ

(b)

Table 2. a) Initial probability distribution for all variables V_j. b) The corresponding *CIM*. The values of k range over all possible numbers of infected neighbors; for a $2D$ torus graph, in the interval $[0; 4]$.

3.2 The equivalent *GSPN* model

GSPN have two different sets of transitions, namely temporal, with an exponentially distributed delay, and immediate transitions, without any delay, the latter having priority over the former. According to the conversion rules described in [9], each node V_j of the *CTBN* (Figure 3) can be converted into the *GSPN* shown in Figure 4. The variable V_j is mapped to place V_j[1], and the value of the variable is mapped into the marking (number of tokens) of the corresponding place. In particular, the marking of the place V_j can be equal to 1, 2, or 3, the same values that the variable V can assume in the *CTBN*. They correspond to the states S, I, R, respectively.

The initialization of V_j is modelled by the immediate transitions TIn_1, TIn_2, and TIn_3. Such transitions are all initially enabled to fire by the place V_j_init, with the effect of setting the initial marking of the place V to 1, 2, or 3. The probability of these transitions to fire corresponds to the initial probability distribution of the variable V_j (Table 2.a).

The variation of the marking of the place V_j is determined by the timed transitions $TSet_1$, $TSet_2$, and $TSet_3$. The transition $TSet_1$ is enabled to fire when the place V_j contains three tokens; the effect of its firing is setting the marking of V_j to 1. Therefore $TSet_1$ represents the state changing from R to S. The transition $TSet_2$ instead, can fire when the marking of the place V_j is

[1] To improve readability in Figure 4 the subscripts that specify the position in the torus graph of the places and transitions are omitted.

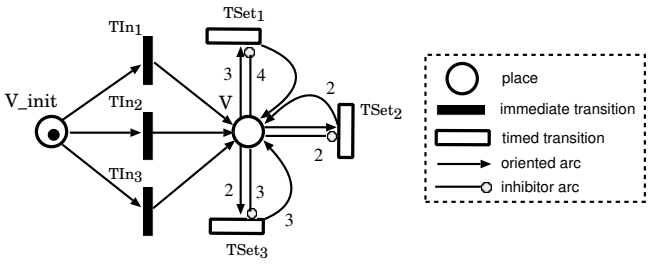

Fig. 4. The *GSPN* corresponding to a node of the *CTBN* of the *SIR* model.

equal to 1 (state S), and turns it to 2 (state I). Finally $TSet_3$ can fire when the marking of V is 2 (I), and changes it to 3 (R).

The conversion of the complete *CTBN* generates a *GSPN* composed by as many instances of the model in Figure 4 as the number of nodes in the *CTBN*. In the *CTBN* a transition rate of a variable can depend on the values of the parent variables representing the neighbour nodes in the case study. In the equivalent *GSPN* the corresponding firing rate depends on the markings of the places representing the parent variables. In Figure 4, the firing rates of $TSet_1$ and $TSet_3$ are constant; the firing rate of $TSet_2$ instead, is marking-dependent, and in particular it changes according to the number of "parent" places V_i ($i \neq j$) currently containing 2 tokens (i.e. the number of infected neighbours). This is expressed in Table 2.b.

3.3 The mean field based model

Let us define $\{\hat{X}_i(\mathbf{v}, t)\}$, with $i \in \{S, I, R\}$ and $\mathbf{v}$ the vertex of the $2D$ torus graph, the mean field approximated process of the model described at the begin of Section 3. The rates of transitions $I \to R$ and $R \to S$ are defined as in the original model, whereas the time and location dependent rate $\chi(\mathbf{v}, t)$ of the transition $S \to I$ of the node $\mathbf{v}$ is defined as:

$$\chi(\mathbf{v}, t) = \lambda + \rho \left(\sum_{\mathbf{v}' \in Neigh(\mathbf{v})} \hat{X}_I(\mathbf{v}, t) \right) \tag{2}$$

where $Neigh(\mathbf{v})$ is the set of the nearest neighbors of the node $\mathbf{v}$. Thus, differently from the *CTBN* model, this rate does not depend on the current exact number of infected neighbors, but on the sum of probabilities that they are infected. The dynamic of the whole process is described by a collection of N 3-by-3 matrices $\mathbf{Q}(\mathbf{v}, t)$ and can be computed by solving the following system of coupled nonlinear non-homogeneous differential equations:

$$\hat{X}_i(\mathbf{v}, 0) = X_i(\mathbf{v}, 0) \tag{3}$$

$$\frac{d\hat{X}_i(\mathbf{v}, t)}{dt} = \hat{X}_i(\mathbf{v}, t)\mathbf{Q}(\mathbf{v}, t). \tag{4}$$

In this way, the construction of the whole state space is avoided by locally solving with standard numerical techniques the process in each node and taking into account the dependencies between neighbors through the mean field approximation. The approach was used to analyze the Markovian agent models [3] and can be seen as a variation of the *fast simulation* technique proposed by Le Boudec in [6] where the process is solved by analysis, instead of stochastic simulation.

4 Inference

Standard inference tasks in a temporal probabilistic model are *prediction* and *smoothing* [14]. Let X_t be a set of variables at time t, and $y_{a:b}$ any stream of observations from time point a to time point b (i.e. a set of instantiated variables). *Prediction* is the task of computing $P(X_{t+h}|y_{1:t})$ for some horizon $h > 0$, i.e. predicting a future state taking into consideration the observation up to now (a special case occurs when $h = 0$, and is called *Filtering*). *Smoothing* is the task of computing $P(X_{t-l}|y_{1:t})$ for some $l < t$, i.e. estimating what happened l time points in the past, given all the evidence (observations) up to now.

Such tasks can be accomplished, depending on the model adopted, by inference procedures usually based on specific adaptation of standard algorithms for *BN*. For instance, in *DBN* models, both exact algorithms based on junction tree [14] as well as approximate algorithms exploiting the net structure [7] or based on stochastic simulation can be employed. In case of *CTBN*, exact inference may often be impractical, so approximations through message-passing algorithms on cluster graphs [17, 20], or through sampling [11, 10] have been proposed.

In the present work, we take advantage of the possibility to map a *CTBN* into a *GSPN*, in order to apply the inference algorithms based on *GSPN* solution algorithms [9]. For example, computing the probability of a given variable assignment $X = x_i$ at time t, will correspond to compute the probability of having i tokens in the place modeling X at time t. Conversion rules from *CTBN* to *GSPN* are described in [9]; an example of conversion is reported in Section 3. For the sake of brevity, we concentrate on the prediction task.

4.1 Prediction Inference

The task of prediction consists in computing the posterior probability at time t of a set of queried variables Q, given a stream of observations (evidence) $e_{t_1}, \ldots e_{t_k}$ from time t_1 to time t_k with $t_1 < \ldots t_k < t$. Every evidence e_{t_j} consists of a (possibly different) set of instantiated variables.

Prediction can then be implemented by repeated transient solution of the corresponding *GSPN* (or *MFM*) at the different times corresponding to the observations and the query. Of course, any observation will condition the evolution of the model, so the suitable conditioning operations must be performed before a new *GSPN* resolution.

Let $Pr\{E\}$ be the probability of the event E, computed from the resulting distribution of a *GSPN* transient (for instance given a variable X, $P(X = x) =$

Procedure PREDICTION
INPUT: a set of queried variables Q, a query time t, a set of temporally labeled evidences $e_{t_1}, \ldots e_{t_k}$ with $t_1 < \ldots t_k < t$
OUTPUT: $P(Q_t | e_{t_1}, \ldots e_{t_k})$

```
- let t_0 = 0;
for i = 1 to k {
      - solve the GSPN transient at time (t_i − t_{i−1});
      - compute from transient, p_i(j) = Pr{X_j | e_{t_i}}
          for X_j ∈ V;
      - update the weights of the immediate ''init''
          transitions of X_j according to p_i(j);
}
- solve the GSPN transient at time (t − t_k);
- compute from transient, r = Pr{Q};
- output r;
```

Fig. 5. The prediction inference procedure ("Init" transitions establish the initial markings of places. In the *GSPN* model in Figure 4, they are: TIn_1, TIn_2, TIn_3.).

$Pr\{\#X' = x\}$ where $\#X'$ is the number of tokens in place X' corresponding to X). The pseudo-code for the prediction procedure is shown in Figure 5.

Notice that, in the special case of *filtering*, the last evidence would be available at the query time (i.e. $t = t_k$ in Figure 5); in such a case, the update of the transition weights (last statement in the `for` cycle) is not necessary, as well as the final transient solution. The procedure would then simply output $Pr\{Q|e_t\}$ computed from the last transient analysis.

5 Results

Let us start by computing the state probability trends of the node in location $\mathbf{v}_0 = (0; 0)$, i.e. the upper-left corner of the grid, assuming all nodes start in state S at time 0. In Figure 6 the results computed by solving the *GSPN* are compared with the *MFM* for two configurations of the parameters: $C_1 := \{\alpha = 0.1,\ \beta = 0.1,\ \gamma = 0.01,\ \rho = 0.05\}$ and $C_2 := \{\alpha = 0.1,\ \beta = 0.1,\ \gamma = 0.1,\ \rho = 0.05\}$ h^{-1}. The model with $N = 9$ nodes has a global state space of $4^9 = 262144$ elements, however increasing the grid raises the size of the global state space at $4^{16} = 4294967296$, an almost intractable dimension for the analysis of the *GSPN* model. Therefore the computations were limited to $N = 9$ and were performed on a laptop equipped with an Intel Core $i5 - 2450M$ CPU at 2.5 GHz, 3 MB SmartCache and $6GB$ RAM, and took a few seconds to solve both the *GSPN* and *MFM*.

We can observe that the accuracy of the mean field approximation in Figure 6.a is good except for discrepancies in the transient phase during the time interval $[5; 15]$ with a maximum absolute error of $\epsilon_t = 0.054423$ and a a maximum absolute error in steady state of $\epsilon_s = 0.0050022$. The approximation achieved

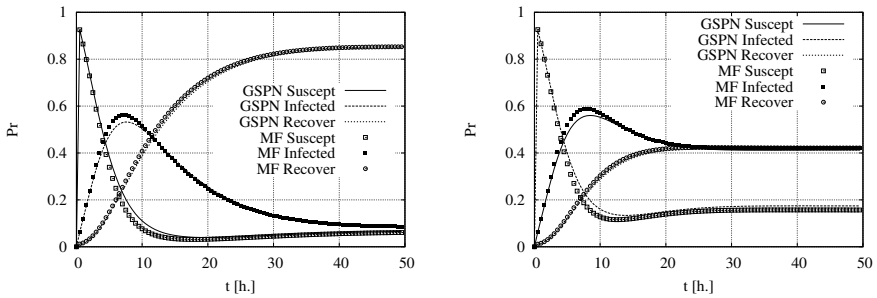

Fig. 6. Comparison between *CTBN* and *MFM* of the infection propagation with parameter configurations: a) C_1; b) C_2.

with configuration C_2 and shown in Figure 6.b is less satisfactory with a maximum absolute error in transient phase of $\epsilon_t = 0.053713$, but in steady state of $\epsilon_s = 0.017347$.

Indeed, the accuracy achieved by the mean field approach depends on the type of the model and the values of its parameters. Let us analyze the queuing model studied in [4]. The authors consider a *CTMC* representing N identical servers, each of them with a single-place queue, and where incoming jobs are directed to the server with the shortest queue. Moreover the arrivals are suspended when all servers are full. Using the notation in [5], the state of the model can be described by three variables X_0, X_1 and X_2 representing the number of servers with 0, 1 and 2 jobs, respectively. We assume that the arrival rate scales with N, thus it is $N \cdot \lambda$, with a service rate of ρ for each queue. Four transitions are possible, in particular:

- incoming job directed to an idle server at rate $N\lambda \cdot \mathbf{1}\{X_0 > 0\}$;
- incoming job directed to a busy server with an empty queue at rate $N\lambda \cdot \mathbf{1}\{X_0 == 0\} \cdot \mathbf{1}\{X_1 > 0\}$;
- processing of a job from a server with an empty queue at rate $\rho \cdot X_1$;
- processing of a job from a server with a full queue at rate $\rho \cdot X_2$

where $\mathbf{1}\{cond\}$ is an indicator function equal to 1 when *cond* is true, 0 otherwise.

As highlighted in [5], the shortest queue policy modeled by the indicator functions yields discontinuities when there are no idle servers (i.e. $X_0 = 0$) and no busy servers with an empty buffer (i.e. $X_1 = 0$). If we consider the model normalized with respect to the system size N, i.e. $\hat{X}_i = X_i/N$ with $i \leq 3$, we have that for initial conditions close to the point $(0, 0, 1)$ the mean field convergence does not hold. In Figure 7 we can observe the behavior of the equivalent $GSPN$ model and the mean field approximation with $N = 50$ servers and parameters $\lambda = 1.5\ h^{-1}$, $\mu = 2\ h^{-1}$. With an initial condition far from the point $(0, 0, 1)$, such

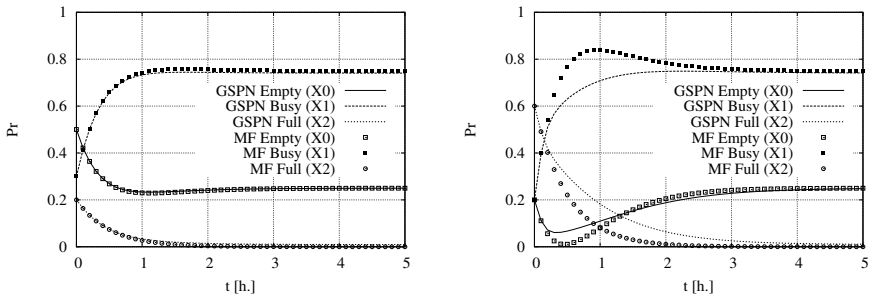

Fig. 7. Comparison between *CTBN* and *MFM* of the queuing model with parameters $\lambda = 1.5$, $\mu = 2$ and initial conditions: a) $\hat{X}_0(0) = 0.5$, $\hat{X}_1(0) = 0.3$, $\hat{X}_2(0) = 0.2$; b) $\hat{X}_0(0) = 0.2$, $\hat{X}_1(0) = 0.2$, $\hat{X}_2(0) = 0.6$.

as $(0.5, 0.3, 0.2)$, the accuracy of the mean field approximation is satisfactory (see Figure 7.a). However, for an initial condition $(0.2, 0.2, 0.6)$ near the critical point, the mean field approximation provides quite different results with a maximum absolute error in the transient phase of $\epsilon_t = 0.124753$.

Time [h]	$Pr_{\mathbf{v}_0}\{S\|ev\}$	$Pr_{\mathbf{v}_0}\{I\|ev\}$	$Pr_{\mathbf{v}_0}\{R\|ev\}$	$Pr_{\mathbf{v}_1}\{S\|ev\}$	$Pr_{\mathbf{v}_1}\{I\|ev\}$	$Pr_{\mathbf{v}_1}\{R\|ev\}$
0	1	0	0	1	0	0
3	0.668589	0.277077	0.054324	0	1	0
4	0.502876	0.408654	0.088464	0.008766	0.888542	0.102685
5	0.385592	0.482107	0.132290	0.008507	0.805611	0.185871

Table 3. Probabilities for prediction inference computed with the *GSPN* model (*ev* is the current accumulated evidence).

Time [h]	$Pr_{\mathbf{v}_0}\{S\|ev\}$	$Pr_{\mathbf{v}_0}\{I\|ev\}$	$Pr_{\mathbf{v}_0}\{R\|ev\}$	$Pr_{\mathbf{v}_1}\{S\|ev\}$	$Pr_{\mathbf{v}_1}\{I\|ev\}$	$Pr_{\mathbf{v}_1}\{R\|ev\}$
0	1	0	0	1	0	0
3	0.582287	0.353869	0.063844	0	1	0
4	0.410119	0.484022	0.105859	0.000441	0.903973	0.095586
5	0.286190	0.556308	0.181044	0.001519	0.817438	0.157502

Table 4. Probabilities for prediction inference computed with the *MFM* (*ev* is the current accumulated evidence).

Concerning prediction inference, let us focus on the *SIR* model with configuration C_1 and assume that at time $t_0 = 0\ h$. All nodes are in a susceptible state and then at time $t_1 = 3\ h$, we observe the nodes in locations $\{(0;1),(1;0),(1;1)\}$ infected. Using the procedure described in Section 4.1 we can compute the state probabilities of the node in the upper-left corner $\mathbf{v}_0 = (0;0)$ and its right neigh-

bor $\mathbf{v}_1 = (0; 1)$ at time $t_2 = 4,5\ h$, given the evidence. The inference is performed by the analysis of both the *GSPN* and the *MFM* and the results are shown in Table 3 and 4, respectively. The results show a quite good match with a maximum absolute error between corresponding values of $\epsilon_t = 0.099402$.

6 Conclusions

In this paper, we investigated the use of the mean field technique to analyze *CTBN*, with the goal of computing the probability distribution of a subset of variables in absence and in presence of evidence. In particular, we applied mean field approximation to a well-known epidemic case study; the results have been compared with *GSPN* analysis output, another way to deal with *CTBN*, obtaining a satisfactory accuracy.

Acknowledgments

This work is original and has a financial support of the *Università del Piemonte Orientale*.

References

1. M. Ajmone-Marsan, G. Balbo, G. Conte, S. Donatelli, and G. Franceschinis. *Modelling with Generalized Stochastic Petri Nets*. John Wiley & Sons, 1994.
2. M. Benaim and J.Y. Le Boudec. A class of mean field interaction models for computer and communication systems. *Performance Evaluation*, 65(11):823 – 838, 2008.
3. A. Bobbio, D. Cerotti, M. Gribaudo, M. Iacono, and D. Manini. *Markovian Agent Models: A Dynamic Population of Interdependent Markovian Agents*, pages 185–203. Seminal Contributions to Modelling and Simulation: 30 Years of the European Council of Modelling and Simulation. Springer, 2016.
4. A. Bobbio, M. Gribaudo, and M. Telek. Analysis of large scale interacting systems by mean field method. In *International Conference on the Quantitative Evaluation of Systems*, 2008.
5. Luca Bortolussi, Jane Hillston, Diego Latella, and Mieke Massink. Continuous approximation of collective system behaviour: A tutorial. *Performance Evaluation*, 70(5):317 – 349, 2013.
6. J. Y. L. Boudec, D. McDonald, and J. Mundinger. A generic mean field convergence result for systems of interacting objects. In *International Conference on the Quantitative Evaluation of Systems*, pages 3–18, September 2007.
7. X. Boyen and D. Koller. Tractable inference for complex stochastic processes. In *Conference on Uncertainty in Artificial Intelligence*, pages 33–42, 1998.
8. J.W. Cho, J.Y. Le Boudec, and Y. Jang. On the validity of the fixed point equation and decoupling assumption for analyzing the 802.11 mac protocol. *Performance Evaluation Review*, 38(2):36–38, October 2010.
9. D. Codetta-Raiteri and L. Portinale. A Petri net-based tool for the analysis of generalized continuous time Bayesian networks. In *Theory and Application of Multi-Formalism Modeling*, pages 118–143. IGI Global, 2013.

10. T. El-Hay, N. Friedman, and R. Kupferman. Gibbs sampling in factorized continuous time Markov processes. In *Conference on Uncertainty in Artificial Intelligence*, 2008.
11. Y. Fan and C. Shelton. Sampling for approximate inference in continuous time Bayesian networks. In *International Symposium on AI and Mathematics*, 2008.
12. K. Gopalratnam, H. Kautz, and D.S. Weld. Extending continuous time bayesian networks. In *National Conference on Artificial Intellicence*, pages 981–986, Pittsburgh, PA, 2005. AAAI.
13. S.L. Lauritzen and T.S. Richardson. Chain graph models and their causal interpretations. *Journal Of The Royal Statistical Society Series B*, 64(3):321–348, 2002.
14. K. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD Thesis, UC Berkley, 2002.
15. U. Nodelman, D. Koller, and C.R. Shelton. Expectation propagation for continuous time bayesian networks. *Computing Research Repository*, abs/1207.1401, 2012.
16. U. Nodelman, C. R. Shelton, and D. Koller. Continuous Time Bayesian Networks. In *Conference on Uncertainty in Artificial Intelligence*, pages 378–387, 2002.
17. U. Nodelman, C. R. Shelton, and D. Koller. Expectation propagation for continuous time Bayesian networks. In *Conference on Uncertainty in Artificial Intelligence*, pages 431–440, 2005.
18. M. Opper and D. Saad, editors. *Advanced mean field methods: theory and practice*. MIT press, 10 2002.
19. L. Portinale, A. Bobbio, D. Codetta-Raiteri, and S. Montani. Compiling dynamic fault trees into dynamic Bayesian nets for reliability analysis: the Radyban tool. In *Bayesian Modeling Applications Workshop*, volume 268 of *CEUR Workshop Proceedings*, Vancouver, Canada, July 2007.
20. S. Saria, U. Nodelman, and D. Koller. Reasoning at the right time granularity. In *Conference on Uncertainty in Artificial Intelligence*, pages 421–430, 2007.
21. A.S. Sznitman. *Topics in propagation of chaos*, pages 165–251. Lecture Notes in Mathematics. Springer, 1991.