Analysis of Stochastic Automata Networks using copula functions

Conference Paper · November 2010

DOI: 10.1109/ALLERTON.2010.5707121 · Source: IEEE Xplore

2 authors:

Amit Surana
United Technologies Research Center
89 PUBLICATIONS **2,026** CITATIONS

SEE PROFILE

Alessandro Pinto
United Technologies Research Center
62 PUBLICATIONS **1,286** CITATIONS

SEE PROFILE

Analysis of Stochastic Automata Networks using Copula Functions

Amit Surana and Alessandro Pinto

***Abstract*— In this paper we develop a copula based approximation framework for scalable analysis of Stochastic Automata Networks (SAN) arising in reliability analysis, and can be described by CTMCs. Copulas provide a general approach to model joint distributions in terms of their marginals. Using copulas functions, the dependencies between the interacting automata in the SAN can be captured in terms of local state probabilities associated with the automata involved, avoiding the need of reachability analysis, which is cursed with state space explosion. We prove results related to invariance of copula with system parameters, and consistency of the approximation. We also outline an empirical procedure for determining copulas that can best represent the underlying dependence in a given SAN. We illustrate this approach through various examples of increasing complexity.**

I. Introduction

Performance and reliability analysis has long relied on Markov Chain models[3]. Given a Continuous Time Markov Chain (CTMC), analysis entails solving the set of differential equations $\dot{\pi} = Q\pi$, where Q is the infinitesimal generator of the CTMC. It is never the case that complex systems are directly modeled using a CTMC. Typically, the system is described using higher level languages, such as Generalized Stochastic Petri Nets (GSPN) [9], that support some sort of compact representation by means of synchronization and composition. The system is then given as a set of sub-systems interacting through some communication primitives. The analysis of these systems starts by first computing the set of reachable system states and corresponding transitions, and then generating a single Markov Chain (see e.g. [4]). Because in the worst case the number of reachable states is the product of the number of states of each sub-system, the reachability analysis step suffers from the well known state explosion problem. After reachability analysis, approximate techniques are available to solve large Markov models (see e.g. [2]). Ideally, the generation of the reachable state space should be avoided. However, reachability analysis is essential to compute performance and reliability metrics. To clarify this point, we start by defining the modeling language that we use, and by providing an example system.

In this article we focus on a restricted form of Stochastic Automata Networks (SAN)[14] that we will call `san`–little SAN. A `san` is pair $\int = (\{\mathscr{A}^{(i)}\}_{i=1\ldots N}, \iff)$. $\mathscr{A}^{(i)}(\mathscr{S}^{(i)}, \Gamma^{(i)}, s_0, \lambda^{(i)}, G^{(i)})$ is a stochastic automaton where $\mathscr{S}^{(i)}$ is a set of states, $\Gamma^{(i)} \subseteq \mathscr{S}^{(i)} \times \mathscr{S}^{(i)}$ is a set of transitions, $s_0 \in \mathscr{S}^{(i)}$ is the initial state, $\lambda^{(i)} : \Gamma^{(i)} \to \mathbb{R}_{\geq 0}$ is a function that associates a rate to each transition, and $G^{(i)} : \Gamma^{(i)} \to \wp(\mathscr{S})$ is a function that associates a guard condition to each transition, where $\wp(\mathscr{S})$ is the power set of $\mathscr{S}$. In this definition, $\mathscr{S} = \times_{i=1\ldots N}\mathscr{S}^{(i)}$ and $\Gamma = \cup_{i=1\ldots N}\Gamma^{(i)}$. The equivalence relation $\iff \subseteq \Gamma \times \Gamma$ captures synchronization among transitions. The semantics of a `san` is rather intuitive and we will only clarify the meaning of synchronization and guard conditions. Two transitions $\gamma^{(i)} = (k_1^i, k_2^i), \gamma^{(j)} = (k_1^i, k_2^i) \in \Gamma \times \Gamma$ such that $\gamma^{(i)} \iff \gamma^{(j)}$ or equivalently represented as

$$\gamma^{(i)}(k_1^i, k_2^i) \iff \gamma^{(j)}(k_1^i, k_2^i)$$

must always occur together which also implies that their rate must be the same. Finally, a transition $\gamma \in \Gamma^{(i)}$ has rate $\lambda^{(i)}(\gamma)$ if the current state of the system belongs to $G^{(i)}(\gamma)$, otherwise the transition is disabled i.e. $\lambda^{(i)}(\gamma) = 0$.

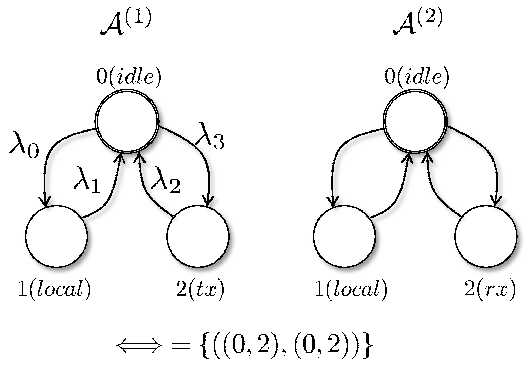

Fig. 1. A simple example of a producer and a consumer.

Figure 1 shows a simple example of a producer $\mathscr{A}^{(1)}$, and a consumer $\mathscr{A}^{(2)}$. Both the producer and the consumer have local tasks (state 1). When the producer wants to transmit something to the consumer, the consumer must receive it. Thus, transition $(0,2)$ of the producer automaton must be synchronized with the transition $(0,2)$ of the consumer automaton.

The equation that describe the transient evolution of the probability that the producer is in state 0 is following:

$$\dot{\pi_0^{(1)}} = -\lambda_0\pi_0^{(1)} - \lambda_3\pi_{00}^{(12)} + \lambda_2\pi_0^{(2)} + \lambda_1\pi_0^{(1)} \qquad (1)$$

where $\pi_{00}^{(12)}$ is the probability that the producer and the consumer are both in state 0. Unfortunately, such joint probabilities are in general not easy to compute (non-product form solution) and one has to typically resort to the computation of set of reachable states by solving the reachability problem incurring in the state space explosion. Some approximate

This work was supported by UTRC.

Amit Surana and Alessandro Pinto are with the United Technologies Research Center: SuranaA@utrc.utc.com, PintoA@utrc.utc.com

methods exists to avoid reachability analysis such as the work in[5], [20], that leverage the properties of the description language. Decomposition is also the basic idea presented in this article. Our work is more general as we model the joint probability using particular parametric functions known as copulas, and learn the value of the parameters through simulations. Thus, this method is easy to automate and can be applied to a wide variety of stochastic models (including GSPN). This method reduces the size of the system of differential equations to be at most $\sum_{i=1\ldots N}|\mathscr{S}^{(i)}|$ which is much smaller than the worst case product space resulting from the reachability analysis. The price that we pay is that the performance metrics computed by this method are local (i.e. relative to each automaton). However, this is not a limitation since the same method (i.e. use of parametric functions) can also be employed to compute the joint probability of any other set of states.

The rest of the paper is structured as follows. In Section II we describe copula based approximation framework, and discuss results related to invariance of copula with system parameters, time invariance of copulas and consistency of the approximation. In Section III, we describe the copula fitting procedure in context of our application for modeling san. In Section IV, we illustrate the copula based approximation framework through various examples of increasing complexity. Finally, in section V we summarize the main results of this paper, and present some future research directions.

II. Approximate compositional modeling

The joint system probability vector $\pi \in \mathbb{R}^{N_o}$ for san where N_o is the number of reachable states, satisfies:

$$\dot{\pi} = R\pi \tag{2}$$

where, $R = Q^T$ and Q is the infinitesimal generator of the CTMC associated with the reachability graph of san. Let $\pi^{(i)} \in [0,1]^{N_i}$ denote the probability vector over states $\mathscr{S}^{(i)}$ for the $i-$th subsystem, where $N_i = |\mathscr{S}^{(i)}|$. There exists a linear transformation $M^{(i)}$ (independent of time and system rate parameters) such that

$$\pi^{(i)} = M^{(i)}\pi, \tag{3}$$

leading to

$$\dot{\pi}^{(i)} \quad = \quad M^{(i)}R\pi = R^{(i)}\pi^{(i)} + L^{(i)}\pi \tag{4}$$

where, $L^{(i)} = (M^{(i)}R - R^{(i)}M^{(i)})$ and $R^{(i)}$ is the transpose of the infinitesimal generator corresponding to the transition $\Gamma^{(i)}$ *not involved* in any synchronization or guard condition. Note that guard condition can be treated as a unidirectional synchronization. The second term in equation (4) arises due to synchronizations/guard conditions, and hence can be expressed entirely in terms of joint probabilities associated with such transitions. The $n-$ synchronization will be identified by the pair of vectors of indices $(\mathbf{i}^n, \mathbf{k}^n)$, where $\mathbf{i}^n = \{i_1, \cdots, i_n\}$ and $\mathbf{k}^n = \{k_1, \cdots, k_n\}$, such that

$$\forall i_m \in \mathbf{i}^n, \exists i_l \in \mathbf{i}^n, i_l \neq i_m, \text{ s.t. } \gamma^{(i_m)}(k_m^{(i_m)}, \cdot) \iff \gamma^{(i_l)}(k_l^{(i_l)}, \cdot) \tag{5}$$

Let Γ_S be a set of all synchronized transitions (including the guard conditions) in the system, i.e

$$\Gamma_S = \{(\mathbf{i}^m, \mathbf{k}^m) : m \in \mathbb{N}, (\mathbf{i}^m, \mathbf{k}^m) \text{satisfy (5)}\}. \tag{6}$$

We shall denote by $\pi_{\mathbf{k}}^{\mathbf{i}}$ the joint probability that the synchronized transitions are executed together, and by

$$\Pi_s = \{\pi_{\mathbf{k}_1}^{(\mathbf{i}_1)}, \cdots, \pi_{\mathbf{k}_{N_s}}^{(\mathbf{i}_{N_s})}\} = \{\pi_{\mathbf{k}}^{\mathbf{i}} : (\mathbf{i}, \mathbf{k}) \in \Gamma_S\}, \tag{7}$$

the set of joint probability associated with all synchronized transitions in the system, where $N_s = |T_S|$. Then system (4) can be rewritten as

$$\dot{\pi}^{(i)} = R^{(i)}\pi^{(i)} + R_s^{(i)}\pi_s^{(i)}, \tag{8}$$

where, $\pi_s^{(i)} \in [0,1]^{N_s^i}$ is the vector of elements in $\Pi_s^{(i)} \subseteq \Pi_s$, which is the set of synchronized transitions affecting the $i-$th system with $N_s^i = |\Pi_s^{(i)}|$, and $R_s^{(i)}$ is the transpose of the infinite generator corresponding to these synchronized transitions. Again, note that the function $G^{(i)}$ automatically takes care of guard conditions (unidirectional synchronizations) by setting of the appropriate rate entries in $R_s^{(i)}$ to be zero.

We seek an approximation of (8), which does not require computations directly involving π_s. Such an approximation can be obtained by replacing π_s by a function of the form

$$\pi_s(t) = \begin{pmatrix} \pi_{\mathbf{k}_1}^{(\mathbf{i}_1)}(t) \\ \vdots \\ \pi_{\mathbf{k}_{N_s}}^{(\mathbf{i}_{N_s})}(t) \end{pmatrix} \approx \begin{pmatrix} f_{1t\Lambda}(\hat{\pi}(t)) \\ \vdots \\ f_{N_s t\Lambda}(\hat{\pi}(t)) \end{pmatrix} \tag{9}$$

where, $f_{it\Lambda}(\hat{\pi}), i = 1\cdots, N_s$ are functions (which in principle can explicitly depend on time and all the system rate parameters $\Lambda = \bigcup_{i=1}^{N}\lambda^{(i)}(\Gamma^{(i)})$) to be chosen from an appropriate class, and $\hat{\pi} = (\hat{\pi}^{(1)}, \cdots, \hat{\pi}^{(N)})^T \in \mathbb{R}^D$ is solution of the *approximate system*

$$\begin{aligned} \dot{\hat{\pi}}^{(1)} &= R^{(1)}\hat{\pi} + R_s^{(1)}\mathbf{f}^{(1)}(\hat{\pi}) \\ &\vdots \\ \dot{\hat{\pi}}^{(i)} &= R^{(i)}\hat{\pi} + R_s^{(i)}\mathbf{f}^{(i)}(\hat{\pi}) \\ &\vdots \\ \dot{\hat{\pi}}^{(N)} &= R^{(i)}\hat{\pi} + R_s^{(N)}\mathbf{f}^{(N)}(\hat{\pi}) \end{aligned} \tag{10}$$

with,

$$\mathbf{f}^{(i)} = \{(f_{j_1}, \cdots, f_{j_{N_s^i}})^T : \forall j_l = 1, \cdots, N_s^i, \pi_{\mathbf{k}_{j_l}}^{(\mathbf{i}_{j_l})} \in \Pi_s^{(i)}\}. \tag{11}$$

For any approximation (10) to be consistent, we require that the $\hat{\pi}$ evolves as a probability vector, i.e. for any $\hat{\pi}(0) \in [0,1]^D$, $\hat{\pi}(t) \in [0,1]^D, \forall t$, where $D = \sum_{i=1}^{N} N_i$.

It turns out that there is a natural function class for the above approximation. To motivate this we introduce binary random variables as follows: for any given $t \in \mathbf{R}$, let $S_k^{(i)}(t)$ be a random variable which takes value 0 if subsystem i is in state k with a probability $P(S_k^{(i)}(t) = 1) = \pi_k^{(i)}(t)$. For brevity we will often suppress the dependence of $S_k^{(i)}(t)$ and

$\pi_k^{(i)}(t)$ on time. The cumulative mass function (CMF) for $S_k^{(i)}$ is given by

$$F_{kt}^{(i)}(s) = P(S_k^{(i)}(t) \le s) = \begin{cases} 1 & s \ge 1 \\ 1-\pi_k^{(i)}(t) & 0 \le s < 1 \\ 0 & s < 0 \end{cases} . \tag{12}$$

In this framework any $n-$synchronized transition with $\mathbf{i} = \{i_1, \cdots, i_n\}, \mathbf{k} = \{k_1, \cdots, k_n\}$ can be represented by the random vector $(S_{k_1}^{(i_1)}, \cdots, S_{k_n}^{(i_n)})$ with the probability of synchronization being

$$\pi_s(t) = P(S_{k_1}^{(i_1)}(t) = 1, \cdots, S_{k_n}^{(i_n)}(t) = 1). \tag{13}$$

Therefore, in this setting the approximation function (9) is determined, once the joint CMF $F_{\mathbf{k}t}^{(\mathbf{i})}(s_1, \cdots, s_n)$ of the random vector $(S_{k_1}^{(i_1)}(t), \cdots, S_{k_n}^{(i_n)}(t))$ can be approximated in terms of it's marginals $F_{k_m t}^{(i_m)}, m = 1, \cdots, n$. Given the joint CMF (or cumulative distribution function(CDF)), while finding the marginal CMF (or CFD) is straightforward, the reverse problem is nontrivial. Remarkably, this problem admits a solution in terms of special class of functions called *copulas*. We briefly review copulas in the next section.

A. Review of Copulas

Copula Definition: A $D-$dimensional copula C is a function on a D dimensional unit cube $\mathscr{B} = [0,1]^D$, $C : \mathscr{B} \to [0,1]$, that satisfies

$$C(u_1, u_2, \cdots, u_D) = 0 \quad \text{if atleast one} \quad u_i = 0 \tag{14}$$

$$C(1, 1, \cdots, u_i, \cdots, 1) = u_i. \tag{15}$$

and C is $D-$increasing, i.e for every $\mathbf{u}_1 = (a_1, \cdots, a_D) \in \mathscr{B}$ and $\mathbf{u}_2 = (b_1, \cdots, b_D) \in \mathscr{B}$ such that $a_i < b_i$,

$$\sum_{i_1=1}^{2} \cdots \sum_{i_D=1}^{2} (-1)^{i_1+\cdots+i_n} C(u_{1i_1}, \cdots, u_{ni_D}) \ge 0 \tag{16}$$

where, $u_{j1} = a_j$ and $u_{j2} = b_j$ for all $j \in \{1, \cdots, D\}$ [11],[19]. Next we state the Sklar's theorem (see [17] for details), which laid the foundation for statistical analysis using copulas.

Sklar's Theorem : Consider the random variables $X_1, \cdots, X_D$ with joint CDF F and continuous marginals, $F_i, i = 1, \cdots, D$. Then F has a unique copula representation, i.e.

$$F(x_1, \cdots, x_D) = C(F_1(x_1), \cdots, F_D(x_D)). \tag{17}$$

If $F_i, i = 1, \cdots, D$ are not all continuous (discrete, or mixed continuous-discrete), Sklar's theorem still holds, but the copula C is not guaranteed to be unique. In discrete case, unique copula representation for F exists only on $\text{Ran}(F_1) \cdots \times \cdots \text{Ran}(F_D)$, where $\text{Ran}(\cdot)$ denotes the range. From Sklar's Theorem we see that for a multivariate distribution function, the univariate margins and the multivariate dependence structure can be separated, and the dependence structure can be represented by a copula.

Examples of Copulas: Two popular class of copulas are the Gaussian Copulas and Student t-Couplas which fall in the elliptical class. *Gaussian* copula, the traditional method to model dependence, is most sensitive to the center of the distribution and implies tail independence. On the other hand, the *Student's t* copula assigns more probability to tail events than the Gaussian copula. The other extreme is the *product* copula

$$C_P(u_1, \cdots, u_D) = u_1 \times \cdots \times u_D, \tag{18}$$

which models complete independence of the underlying random variables. The product copula falls in a more general class of *Archimedean* copulas, which have the form

$$C_\phi(u_1, \cdots, u_D) = \phi^{-1}\{\phi(u_1) + \cdots \phi(u_D)\}, \tag{19}$$

where, $\phi : [0,1] \to [0,\infty)$ is a bijection with

$$\phi(1) = 0, \qquad (-1)^i \frac{d^i}{dt^i}\phi^{-1}(t) > 0, \quad \forall i \in \mathbb{N}. \tag{20}$$

ϕ is called a generator, e.g. of which include $\phi(t) = -\log t$ (Product copula, see (18)), $\phi(t) = \frac{t^{-\alpha}-1}{\alpha}, \alpha \in (0,\infty)$ (Clayton copula),$\phi(t) = (-\log t)^\alpha, \alpha \in [1,\infty)$ (Gumbel copula), and $\phi(t) = \frac{e^{\alpha t}-1}{e^\alpha - 1}, \alpha \in \mathbb{R} \setminus \{0\}$ (Frank copula). The generator depends only on a single parameter, and hence an Archimedean copula is completely specified once ϕ and α are given. In bivariate case, there are 22 such families known [19].

Properties of Copulas: Copulas satisfy Frechet Hoeffding inequality

$$\max(\sum_{i=1}^{D} u_i + 1 - D, 0) \le C(u_1, u_2, \cdots, u_D) \le C_m, \tag{21}$$

where,

$$C_m = \min(u_1, u_2, \cdots, u_D), \tag{22}$$

is known as the *maximum* copula and represents perfect positive dependence between the random variables. Copulas are globally Lipsitz, i.e. $\forall \mathbf{u}_1, \mathbf{u}_2 \in [0,1]^n$

$$|C(\mathbf{u}_1) - C(\mathbf{u}_2)| \le ||\mathbf{u}_1 - \mathbf{u}_2||_1. \tag{23}$$

Another nice property of copulas is that for strictly monotone transformations of the random variables, copulas are either invariant, or change in certain simple ways [12].

Copula Density: Assuming copula density

$$c = \frac{\partial^D C}{\partial x_1 \cdots \partial x_D}, \tag{24}$$

exists, the joint density function can be expressed as

$$f(x_1, \ldots, x_D) = c(F_1(x_1), \cdots, F_D(x_D)) \prod_{j=1}^{D} f_j(x_j) \tag{25}$$

where, f_i are univariate density functions. In discrete this takes a different form, for e.g. copula density for two discrete random variables S_1, S_2 with CMF F_i is given by

$$c(F_1(s_1), F_2(s_2)) = C(F_1(s_1), F_2(s_2)) - C(F_1(s_1 - 1), F_2(s_2)) - C(F_1(s_1), F_2(s_2 - 1)) + C(F_1(s_1 - 1), F_2(s_2 - 1)) \tag{26}$$

This formula generalizes to arbitrary higher dimensions [19].

B. Using Copulas in san analysis

Based on Sklar's theorem stated in previous section, for any $n-$ synchronized transition with $\mathbf{i} = \{i_1, \cdots, i_n\}, \mathbf{k} = \{k_1, \cdots, k_n\}$ and for any given time t, the joint CMF $F^{(\mathbf{i})}_{\mathbf{k}t}(s_1, \cdots, s_n)$ of the random vector $(S^{(i_1)}_{k_1}(t), \cdots, S^{(i_n)}_{k_n}(t))$ is linked to its marginals $F^{(i_m)}_{k_m t}, m = 1, \cdots, n$, through a copula C_t as

$$F^{(\mathbf{i})}_{\mathbf{k}t}(s_1, \cdots, s_n) = C_{t\Lambda}(F^{(i_1)}_{k_1 t}(s_1), \cdots, F^{(i_n)}_{k_n t}(s_n)), \qquad (27)$$

where, in principle $C_{t\Lambda}$ can depend on time and the system rate parameters Λ. Once again note that the copula $C_{t\Lambda}$ above is not guaranteed to be unique as the underlying random variables are discrete (in fact binary), though all such copulas agree on $\text{Ran}(F^{(i_1)}_{k_1 t}) \cdots \times \cdots \text{Ran}(F^{(i_n)}_{k_n t}) \equiv \{0, 1 - \pi^{i_1}_{k_1}(t), 1\} \cdots \times \{0, 1 - \pi^{i_n}_{k_n}(t), 1\}$. In particular, for the approximation (9) we only require $\pi^{(\mathbf{i})}_{\mathbf{k}}(t)$, the probability of subsystems $\mathbf{i} = (i_1, \cdots, i_n)$ being jointly in the state $\mathbf{k} = (k_1, \cdots, k_n)$. This can be recovered though the relation

$$\pi^{(\mathbf{i})}_{\mathbf{k}}(t) = c_{t\Lambda}(F^{(i_1)}_{k_1 t}(1), \cdots, F^{(i_n)}_{k_n t}(1)), \qquad (28)$$

where, $c_{t\Lambda}$ is the copula density corresponding to the copula $C_{t\Lambda}$. For e.g., using relation (26) the joint probability that subsystem i_1 is in state k_1, and subsystem i_2 is in state k_2 can be expressed as

$$\begin{aligned} \pi^{(i_1 i_2)}_{k_1 k_2} &= c(F^{(i_1)}_{k_1}(1), F^{(i_2)}_{k_2}(1)) \\ &= 1 - (1 - \pi^{(i)}_{k}) - (1 - \pi^{(j)}_{m}) + C(1 - \pi^{(i)}_{k}, 1 - \pi^{(j)}_{m}) \end{aligned}$$

or

$$\begin{aligned} &\pi^{(i_1 i_2)}_{k_1 k_2}(t) = c_t(\pi^{(i_1)}_{k_1}(t), \pi^{(i_2)}_{k_2}(t)) \\ &= \pi^{(i_1)}_{k_1}(t) + \pi^{(i_2)}_{k_2}(t) + C_{t\Lambda}(1 - \pi^{(i_1)}_{k_1}(t), 1 - \pi^{(i_2)}_{k_2}(t)) - 1. \end{aligned}$$

1) Example I: Approximate Representation: For the system described in section I (see Figure 1), the approximate representation using copula functions takes the form,

$$\begin{aligned} \dot{\pi}^{(1)} &= R^{(1)}\pi^{(1)} + R^{(1)}_s \pi^{(1)}_s \\ \dot{\pi}^{(2)} &= R^{(2)}\pi^{(2)} + R^{(1)}_s \pi^{(2)}_s \end{aligned} \qquad (29)$$

where,

$$R^{(1)} = \begin{pmatrix} -\lambda_0 & \lambda_1 & \lambda_2 \\ \lambda_0 & -\lambda_1 & 0 \\ 0 & 0 & -\lambda_2 \end{pmatrix}, \quad R^{(1)}_s = \begin{pmatrix} -\lambda_3 \\ 0 \\ \lambda_3 \end{pmatrix},$$

$$R^{(2)} = \begin{pmatrix} -\lambda_4 & \lambda_5 & \lambda_6 \\ \lambda_4 & -\lambda_5 & 0 \\ 0 & 0 & -\lambda_6 \end{pmatrix}, \quad R^{(2)}_s = \begin{pmatrix} -\lambda_3 \\ \lambda_3 \\ 0 \end{pmatrix},$$

with $\pi^{(1)}_s = \pi^{(12)}_{00}$, $\pi^{(2)}_s = \pi^{(12)}_{00}$ and

$$\pi^{(12)}_{00} = \pi^{(1)}_0 + \pi^{(2)}_0 + C_{t\Lambda}(1 - \pi^{(1)}_0, 1 - \pi^{(2)}_0) - 1 \qquad (30)$$

is the approximation vector with $\Lambda = \{\lambda_1, \lambda_2, \cdots, \lambda_6\}$.

We next show that the copula $C_{t\Lambda}$ appearing in (27) cannot explicitly depend on time, and is also independent of the system rate parameters. The intuition behind this is that the structure of reachability graph (which is an alterative representation of dependencies in the system) does not change with time or rate parameters. Consequently, the relation (3) which gives the marginal probabilities in terms of the joint is independent of time and the rate parameters. As a result, we expect the converse to be true, i.e. the relation (copulas) which give joint in terms of marginals, should be time invariant and independent of parameter values. This is formalized in the next theorem.

Theorem 2.1 (Independence from rates and time): The copulas $C_{t\Lambda}$ appearing in (27) cannot explicitly depend on time, and is independent of the system rate parameters Λ.

The proof can be found in [13]. This is a very powerful result, since once the appropriate copula functions have been determined from system simulation traces for some choice of the rate parameter values, they can be used for analyzing the system behavior for any other set of rate parameters and for any time instant. It is most convenient to determine the best copulas fit by analyzing the steady state behavior of the system for nominal rate parameters. This can be accomplished by simulating the entire system using the standard Monte Carlo techniques (see section III for details).

Motivated by this, we will take the approximation functions in (9) to be,

$$\begin{pmatrix} f_1(\hat{\pi}) \\ \vdots \\ f_{N_s}(\hat{\pi}) \end{pmatrix} = \begin{pmatrix} c_1(\hat{\pi}^{(i^1_1)}_{k^1_1}, \cdots, \hat{\pi}^{(i^1_{N_1})}_{k^1_{N_1}}) \\ \vdots \\ c_{N_s}(\hat{\pi}^{(i^{N_s}_1)}_{k^{N_s}_1}, \cdots, \hat{\pi}^{(i^{N_s}_{N_{N_s}})}_{k^{N_s}_{N_{N_s}}}) \end{pmatrix}, \qquad (31)$$

where,

$$c(\pi^{(i_1)}_{k_1}, \cdots, \pi^{(i_n)}_{k_n}) \equiv c(F^{(i_1)}_{k_1}(1), \cdots, F^{(i_n)}_{k_n}(1)). \qquad (32)$$

Final point to note is that, while the existence of a copula which represents the joint CMF is guaranteed, there is no constructive way to determine it exactly. In practise one can only determine a copula (from a finite class of chosen copulas) which can *best* capture the dependency structure (see section III-B for details). The next theorem guarantees that despite any numerical error in estimating the copulas underlying the synchronized transitions, the approximate system (10) is always well behaved.

Theorem 2.2 (Consistency): The solutions of the approximate system (10) with the approximation vector (31) is well behaved, i.e. for any $\hat{\pi}(0) \in [0,1]^D$,

$$\sum_{j=1}^{N_i} \hat{\pi}^{(i)}_j(t) = l, \quad \forall t \in \mathbb{R}, \qquad \forall i = 1, \cdots, N. \qquad (33)$$

For proof see [13].

III. Solution of the Underlying CTMC using the Approximation Framework

A. Simulator

In order to capture dependencies in the system with copulas, the first step is to observe the dependencies through simulations. By simulating the complete san (typically for

nominal rate parameters), one can generate a time trace which describes how the states, the different subsystems are in, evolve over time. The system is to be simulated for long enough time so that a steady state is reached. From this trace, one can extract a subsequence corresponding to each synchronized transition. We shall denote by

$$\mathbf{s}^{\mathbf{i}}_{\mathbf{k},n_1} = (s^{(i_1)}_{k_1,n_1},\cdots,s^{(i_m)}_{k_m,n_1}),\cdots,\mathbf{s}^{\mathbf{i}}_{\mathbf{k},n_f} = (s^{(i_1)}_{k_1,n_f},\cdots,s^{(i_m)}_{k_m,n_f}), \tag{34}$$

the steady state trace of length $n_f - n_1 + 1$ corresponding to the evolution of states $(S^{(i_1)}_{k_1},\cdots,S^{(i_m)}_{k_m})$ involved in a $m-$ synchronized transition with $\mathbf{i} = \{i_1,\cdots,i_m\}, \mathbf{k} = \{k_1,\cdots,k_m\}$. Given the sampled data from simulations, copula fitting procedure is described in the next section.

B. Copula Fitting

In this section we summarize the procedure for determining copula functions which can most effectively capture the dependence structure in the observed data. There are several steps involved in this process.

1) Test Space Selection: First an appropriate test space needs to be constructed. A copula test space $\mathscr{C}$ is a finite subset of the set consisting of members of all possible copula families from which the best will be chosen for a particular fitting application. In order to construct an appropriate test space, the following aspects should be taken into account[10]: *Size* (i.e. number of distinct copula families in test space), *Diversity* (i.e. distinct properties exhibited by copulas in the test space) and *Relevance* (i.e. only comparable copulas enter the test space). Relevance of copula can be determined based on how effective it is in describing a given dependence structure in terms of its parameter range. Dependency measures such as Kendall's Tau and Spearman's Rho [11] can be used. Consider (X,Y) be distributed according to H, and C be the associated copula. Kendall's Tau which measurers the probabilities of concordance and discordance for two independent pairs (X_1,Y_1) and (X_2,Y_2) each with distribution H, is given by

$$\tau = 4\int_0^1\int_0^1 C(u,v)dC(u,v) - 1. \tag{35}$$

Given bivariate observations $(x_{1i},x_{2i}), i = 1,\cdots,n$ distributed according to H, an empirical estimate of τ can be obtained using

$$\hat{\tau} = \frac{1}{\binom{n}{2}}\sum_{i<j}\text{Sign}[(x_{1i}-x_{1j})(x_{2i}-x_{2j})]. \tag{36}$$

In some instances such as for Archimedean copula (see section II-A), estimate of Kendall's τ, can also be used to obtain an estimate of the parameter α by solving

$$1 + 4\int_0^1 \frac{\phi(t)}{\phi'(t)}dr = \hat{\tau}. \tag{37}$$

For e.g, using this relation, for Clayton copula $\theta = \frac{2\tau}{1-\tau}$, for Gumbel $\theta = \frac{1}{1-\tau}$, and for Morgenstern copula $\theta = \frac{9}{2}\tau$. Next section describes a general approach for copula parameter estimation.

2) Copula Parameter Estimation: For any parametric copula family in the test space $\mathscr{C}$, the value of the parameter (vector) needs to be estimated. Recall, that copulas involve two underlying functions: the marginal CDF and the joint CDF. To estimate copula parameters, the first issue consists in specifying how to estimate separately the marginals and the joint law. Moreover, some of these functions can be fully known. Depending on the assumptions made, some quantities have to be estimated parametrically, or semi parametrically or even non-parametrically. In non-parametric case one can use a completely empirical approach or invoke smoothing methods well-known in statistics: such as kernels,wavelets, orthogonal polynomials, nearest neighbors, etc [15]. In our work, we focus on parametric estimation using the standard Maximum Likelihood Estimation (MLE) approach.

For parametric estimation we assume that copula to be estimated belongs to a family $\{C_\theta, \theta \in \Theta\}$, where Θ is space of parameters. Consider a copula-based parametric model of random vector $X = (X_1,\cdots,X_D)$ with cumulative distribution function:

$$F(\mathbf{x};\alpha_1,\cdots,\alpha_D,\theta) = C(F_1(x_1,\alpha_1),\cdots,F_D(x_D,\alpha_D);\theta), \tag{38}$$

where, F_i are the univariate CDFs with parameters $\alpha_1,\cdots,\alpha_D$, respectively. For a sample of size n with observed vectors $\mathbf{x}_i = (x_{i1},\cdots,x_{iD}), i = 1,\cdots,n$ one can form two types of log-likelihood functions:

$$L_j(\alpha_j) = \sum_{i=1}^n \log f_j(x_{ij};\alpha_j), \qquad j = 1,\cdots,D, \tag{39}$$

and

$$\begin{aligned} L(\alpha_1,\cdots,\alpha_D,\theta) &= \sum_{i=1}^n \log f(\mathbf{x}_i,\alpha_1,\cdots;\alpha_n,\theta) \\ &= \sum_{i=1}^n \log\left(c(F_1(x_{i1};\alpha_1),\cdot,F_D(x_{iD};\alpha_D);\theta)\prod_{j=1}^D f_j(x_{ij};\alpha_j)\right), \end{aligned} \tag{40}$$

where, we have used (25). Based on these likelihood functions, there are three approaches to estimate the parameters [16]:

1. Full Maximum Likelihood (FML) method in which the estimate $\hat{\alpha}_1,\cdots,\hat{\alpha}_D,\hat{\theta}$ are obtained by simultaneously solving (39) and (40).
2. Inference function of Margins (IFM) method, in which the parameter estimates are obtained sequentially: first parameters $\hat{\alpha}_1,\cdots,\hat{\alpha}_D$ are obtained by solving (39), and then used in (40) to determine θ.
3. Canonical Maximum Likelihood (CML) method which is like IFM, but differs in that no assumptions are made about the parametric form of the marginal distributions F_i, i.e. F_i are taken to be empirical CDFs. Hence, only θ remains to be estimated.

In `san` setting with underlying discrete random variables, CML method is a natural approach for copula parameter estimation. In the first step the marginal distribution is estimated, which just involves estimating $\pi_i^{(k)}$ in (12) for

the underlying binary random variable. For this step, MLE is trivial, and is equivalent to estimating π empirically. Given this marginal distribution, in the second step copula dependency parameter is estimated by constructing the log-likelihood function as described above. For simplicity, let the synchronized transition involve the state pair $(S_{k_1}^{i_1}, S_{k_2}^{i_2})$, then the joint probability in terms of the marginal is given in terms of copula density as (see Eq. 26)

$$\begin{aligned}
c(F_{k_1}^{i_1}(0), F_{k_2}^{i_2}(0); \theta) &= C(1-\pi_{k_1}^{i_1}, 1-\pi_{k_2}^{i_2}; \theta) \\
c(F_{k_1}^{i_1}(1), F_{k_2}^{i_2}(0); \theta) &= 1-\pi_{k_2}^{i_2} - C(1-\pi_{k_1}^{i_1}, 1-\pi_{k_2}^{i_2}; \theta) \\
c(F_{k_1}^{i_1}(0), F_{k_2}^{i_2}(1); \theta) &= 1-\pi_{k_1}^{i_1} - C(1-\pi_{k_1}^{i_1}, 1-\pi_{k_2}^{i_2}; \theta) \\
c(F_{k_1}^{i_1}(1), F_{k_2}^{i_2}(1); \theta) &= \pi_{k_1}^{i_1} + \pi_{k_2}^{i_2} + C(1-\pi_{k_1}^{i_1}, 1-\pi_{k_2}^{i_2}; \theta) - 1.
\end{aligned}$$

Using above relations, the copula parameter θ can be obtained using ML estimation

$$\frac{\partial L^n}{\partial \theta} = 0, \tag{41}$$

where,

$$L^n(\theta) = \sum_{i=1}^{n} \log c(F_{k_1}^{i_1}(s_{k_1 i}^{i_1})), F_{k_2}^{i_2}(s_{k_2 i}^{i_2}); \theta) \tag{42}$$

and $(s_{k_1 i}^{i_1}, s_{k_2 i}^{i_2}), i = 1 \cdots, n$ is the binary sequence of system being jointly in state $(S_{k_1}^{i_1}, S_{k_2}^{i_2})$, obtained from Monte Carlo simulations as described previously. This approach generalizes to synchronization involving three or more states in a straightforward manner by using an appropriate generalized formula for copula density, as alluded in the end of section II-A.

For discrete random variables, the ML approach for estimating parameters often suffers from convergence problems. An alternative is to appropriately transform the discrete random variables into continuous and then apply the ML estimation for the resulting continuous random variable [6].

3) Goodness of Fit: Finally, a goodness-of-fit test is required for choosing the best copula in the chosen class $\mathscr{C}$. Depending on the discrete or the continuous setting used for estimating copula parameters, and the type of copula employed for fitting, there exists several tests for determining the goodness of fit [11], [19]. An important class is that of *blanket tests* which requires minimal tuning of the test parameters. It includes tests such as those based on empirical copula, Kendalls transform and Rosenblatts transform [7]. For Archimedean copulas, there is an alternative simpler approach for determining the goodness of fit, see [8], [1]. In discrete setting any of these approaches can be used as well. Alternatively, one can employ a l_1 norm or *Kulber-Leiber* divergence to compare the empirical joint distribution estimated from the simulation trace, and that obtained based on the fitted copula. For simplicity, we describe this approach in detail for the bivariate case. Let

$$\mathbf{s}_{\mathbf{k},n_1}^{\mathbf{i}} = (s_{k_1,n_1}^{(i_1)}, s_{k_2,n_1}^{(i_2)}), \quad \mathbf{s}_{\mathbf{k},n_1+1}^{\mathbf{i}} = (s_{k_1,n_1+1}^{(i_1)}, s_{k_2,n_1+1}^{(i_2)}) \cdots$$
$$\mathbf{s}_{\mathbf{k},n_f}^{\mathbf{i}} = (s_{k_1,n_f}^{(i_1)}, s_{k_2,n_f}^{(i_2)}),$$

be the steady state trace of the pair $(S_{k_1}^{i_1}, S_{k_2}^{i_2})$ involved in of synchronized transition. Let $\hat{h}_{\mathbf{k}}^{\mathbf{i}}(i,j), i = 0,1, j = 0,1$ be the empirical estimates of the joint probability mass based on this trace. Similarly, for a given copula with the estimated parameter (based on this trace), one can obtain an estimate of the joint probability by using (26), which we shall denote by $c_{\mathbf{k}}^{\mathbf{i}}(i,j), i = 0,1, j = 0,1$. Then, one can determine goodness of copula fit using either a l_1 norm

$$l_1(\hat{h}_{\mathbf{k}}^{\mathbf{i}}, c_{\mathbf{k}}^{\mathbf{i}}) = \sum_{i=0}^{1}\sum_{j=0}^{1} |\hat{h}_{\mathbf{k}}^{\mathbf{i}}(i,j) - c_{\mathbf{k}}^{\mathbf{i}}(i,j)| \tag{43}$$

or the *Kulber-Leiber* divergence

$$D(\hat{h}_{\mathbf{k}}^{\mathbf{i}}, c_{\mathbf{k}}^{\mathbf{i}}) = \sum_{i=0}^{1}\sum_{j=0}^{1} \hat{h}_{\mathbf{k}}^{\mathbf{i}}(i,j) \log \frac{\hat{h}_{\mathbf{k}}^{\mathbf{i}}(i,j)}{c_{\mathbf{k}}^{\mathbf{i}}(i,j)}. \tag{44}$$

C. Algorithm

In this section we summarize the overall steps:

(I) Use Monte Carlo Simulations to generate a time traces of steady state evolution of the system (for nominal rate parameters) under consideration. Extract from this the time trace subsequences which correspond to each of the synchronized transition as described in section III-A.
(II) For a given synchronized transition, choose an appropriate test class for fitting copula as discussed in section III-B.1.
(III) Use CML approach to estimate copula parameters as described in III-B.2.
(IV) Determine the best copula from the test space using the procedure given in section III-B.3
(V) Repeat steps II-IV for each synchronized transition.

IV. Numerical results

In this section we apply our methodology to two `san` of increasing complexity. For each case we restrict the test space to be composed entirely of Archimedean copulas: specifically *Product*, *Frank*, *Clayton* and *Gumbel*. These copulas were selected since, they most of them span the entire range of Kendal's Tau. In addition, the maximum copula C_m (see Eq. 22) corresponding to upper bound of Frechet Hoeffding inequality (21) was also added to make the test space comprehensive. We follow the algorithm given in section III-C for copula fitting. Finally, we compare the solution obtained from approximate system (10) with the exact solution obtained using the reachability analysis.

A. Example I

Here we revisit the system described in (II-B.1). Given the time traces from simulation, we learn the dependency between the subsystems using copulas. First we estimate Kendall's tau to be $\tau = 0.0065$, suggesting a weak dependence. The corresponding parameter estimates were found to be: Clayton $\theta = 0.0311$, Frank $\theta = 0.1073$ and Gumbel $\theta = 1.0132$, with Frank copula giving the best fit. We found that the continuous extension method (see [12]) also gives similar results. Figure 2 shows the solution of system (29)

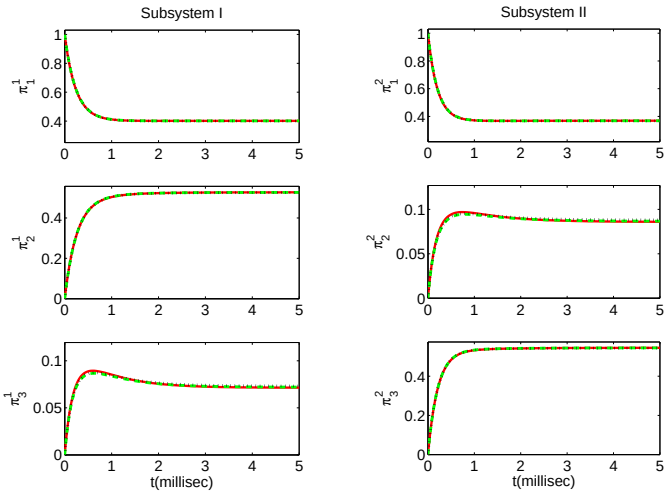

Fig. 2. Comparison of solution obtained using approximation based on independent (green doted) and Frank copulas (black), with that obtained using reachability analysis (red curve). The rates are $\lambda_0 = 2000$, $\lambda_1 = 1524$, $\lambda_2 = 600$, $\lambda_3 = 1234$, $\lambda_4 = 1025$, $\lambda_5 = 2553$, and $\lambda_6 = 1729$.

using Frank copula, compared with the true solution (red curve) obtained by constructing the full reachability graph. Note that the product copula also performs well in capturing dependence (see the green curve in Fig. 2), consistent with the weak dependence pointed out earlier.

B. Example II

In this section we consider a relatively large example involving 11 automata. Figure 3 shows, the different subsystems, synchronized transitions and the guard conditions. There are total 14 synchronized transitions (only a subset shown in figure), 2 guard conditions and 18 distinct rate parameters. Out of 14 synchronized transitions, 2 involve three states, while the remaining ones involve 2 states (see also the first column of I). The local $R^{(i)}$ and synchronized $R_s^{(i)}$ rate matrices, and the corresponding synchronization vector $\pi_s^{(i)}$ for each subsystem can be easily constructed; due to lack of space we do not present them here. The guard conditions are treated as unidirectional synchronized transitions, as was noted in section II. Again we used the test space comprising of following copulas: *Product*,*Frank*,*Clayton*, *Gumbel* and C_m. For each synchronized transition and the guard condition, the best copula from the test space and the corresponding copula parameter is listed in Table I. We have followed a global ordering of the states as depicted in figure 3. Note the following points:

1 The product copula is never selected.
2 For each of the synchronized transition involving three states, C_m is always found to be the best one.
3 For the two guard condition, the best copula is found to be *Frank* with negative parameter value. This suggests that the guard condition induces a negative Kendall's Tau, i.e. a discordance in the dependence.

Finally, figure 4 shows the response of the system (dashed black curves), comparing it with that obtained based on the

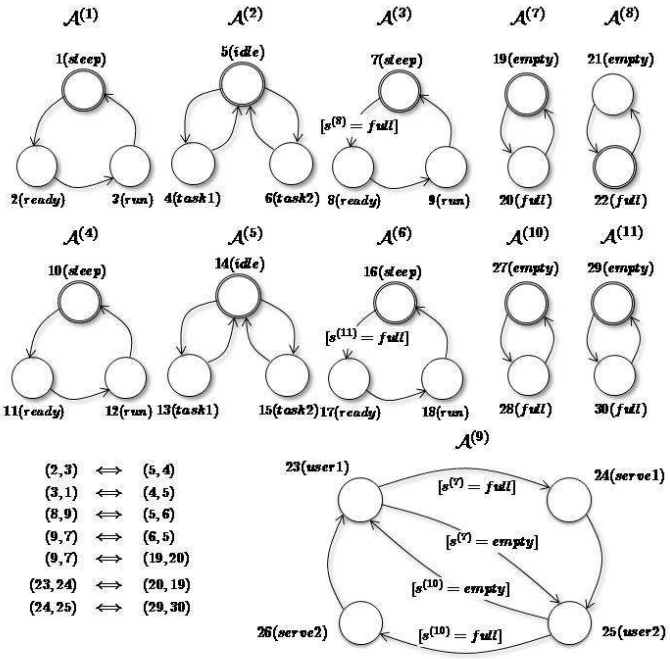

Fig. 3. Figure for larger system

TABLE I
BEST FIT COPULAS FOR EXAMPLE II.

Sync	Copula	θ	Sync	Copula	θ
$\pi_{2,5}^{(1,2)}$	Frank	3.9602	$\pi_{7,22}^{(3,7)}$	Clayton	14.9366
$\pi_{3,4}^{(1,2)}$	C_m	-	$\pi_{20,23}^{(7,9)}$	Clayton	0.4821
$\pi_{5,8}^{(2,3)}$	Frank	0.3059	$\pi_{24,29}^{(9,11)}$	Clayton	50.3630
$\pi_{6,9,19}^{(2,3,7)}$	C_m	-	$\pi_{25,28}^{(9,10)}$	Frank	0.9269
$\pi_{11,14}^{(4,5)}$	Gumbel	1.3605	$\pi_{26,21}^{(9,8)}$	Clayton	47.0802
$\pi_{12,13}^{(4,5)}$	C_m	-	$\pi_{16,30}^{(6,11)}$	Frank	68.9264
$\pi_{17,14}^{(6,5)}$	Frank	0.6456	$\pi_{23,19}^{(9,7)}$ (Guard)	Frank	-1.1682
$\pi_{18,15,27}^{(6,5,10)}$	C_m	-	$\pi_{25,27}^{(9,10)}$(Guard)	Frank	-0.9269

reachability analysis (red solid curves) and that obtained based on assuming that all underlying copulas are of the product form C_P (dashed green curves). Subplots a) and d) correspond to nominal rate parameters, while the subplots c) and d) are for different set of rate values many of which differ from nominal values by a factor of more than 10. For this system, the reachability graph comprises of 150 states compared to the 30 states required in the copula based approximation framework.It is clear from the figure 4 that the copula based framework can accurately capture the system response for wide range of parameter values, while avoiding state space explosion of underlying state space.

V. CONCLUDING REMARKS

In this work we have developed a copula based approximation framework for scalable analysis of stochastic automata networks `san` which can be described by CTMCs. Using copulas functions, the dependencies between the interacting

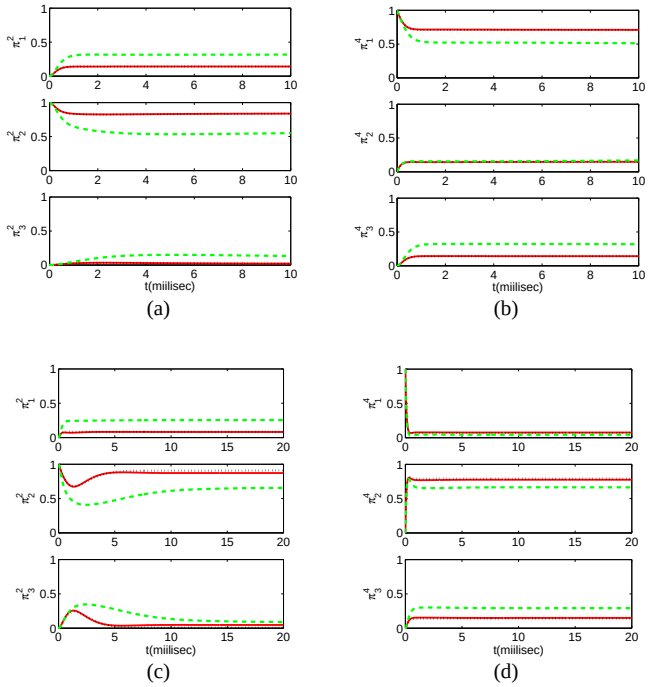

Fig. 4. Comparison of solution (red is ground truth) with all independent copulas (green) and with obtained using copula selection procedure (black) for subsystems $\mathscr{A}^2$ (subplots a) & c)) and $\mathscr{A}^4$ (subplots b) & d)) in the large example. Subplots a) and b) correspond to nominal rate, parameters, while subplots c) and d) are similar plots but for a different set of parameters, many of which differ from nominal values by a factor of more than 10.

stochastic automata subsystems in san can be captured in terms of local state probabilities associated with such automata involved, avoiding the need for the reachability analysis which is cursed with state space explosion. We showed that copula based approximation is consistent, and in principle can capture the true system behavior to any desired level of accuracy. We also proved that the dependency structure captured in form of copulas is time invariant and also is invariant under the change of system rate parameters. Thus, once the appropriate copula functions have been learnt from Monte Carlo simulation of the complete system for nominal rate parameters, the same copulas can be used for analyzing the system with any other set of rate parameters. We described in detail various approaches to learn copulas from a given simulation trace. Finally, we successfully demonstrated the overall copula based approximation framework through various examples of increasing complexity.

There are several possible extensions. In many applications, compositional analysis of petri nets with general sojourn distributions is required. In this case Markov Regererative Processes form the underlying stochastic processes [18], and the process evolution is governed by integro-differential equations. Extending copula framework in this setting will be of great practical interest. Finally, exploring copula based framework for rapid exploration in synthesis problem is another direction to pursue in the future.

VI. Acknowledgements

The funding provided by United Technologies Research Center for this work is greatly appreciated. Authors would like to thank Andrew Lim from the IEOR department at the University of California at Berkeley for suggesting the use of copula functions. They also would like to thank Andrzej Banaszuk at UTRC for fruitful discussions and feedback on this work.

References

[1] P. Barbe, C. Genest, K. Ghoudi, and B. Remillard. On kendall's process. *Journal of Multivariate Analysis*, 58(2):197–229, 1996.

[2] A Bobbio and K S Trivedi. An aggregation technique for the transient analysis of stiff markov chains. *IEEE Trans. Comput.*, 35(9):803–814, 1986.

[3] Gunter Bolch, Stefan Greiner, Hermann de Meer, and Kishor Shridharbhai Trivedi. *Queueing Networks and Markov Chains*. Wiley-Interscience, 2005.

[4] Gianfranco Ciardo, Joshua Gluckman, and David Nicol. Distributed state-space generation of discrete-state stochastic models. *INFORMS Journal of Computing*, 10:82–93, 1996.

[5] Gianfranco Ciardo and Kishor S. Trivedi. A decomposition approach for stochastic reward net models. *Perf. Eval*, 18:37–59, 1993.

[6] M. Denuita and P. Lamberta. Constraints on concordance measures in bivariate discrete data. *Journal of Multivariate Analysis*, 93:40–57, 2005.

[7] C. Genest, B. Remillard, and D. Beaudoin. Goodness-of-fit tests for copulas: A review and a power study. *Insurance: Mathematics and Economics*, 44(2):199–213, 2009.

[8] C. Genest and L. Rivest. Statistical inference procedures for bivariate archimedean copulas. *Journal of the American Statistical Association*, 88:1034–1043, 1993.

[9] G. Conte S. Donatelli M. Ajmone Marsan, G. Balbo and G. Franceschinis. *Modelling with Generalized Stochastic Petri Nets*. John Wiley and Sons, 1995.

[10] F. Michiels and A. Schepper. A copula test space model, how to avoid the worng copula choice. *Kybernetika*, 44(6):864–878, 2008.

[11] R. B. Nelsen. *An introduction to copulas*. Lecture Notes in Statistics. New York: Springer Verlag, 1999.

[12] J. Neslehova. *Dependence of Non-Continuous Random Varaibles*. Ph.D. Thesis. ETH Zurich, Switzerland, 2004.

[13] A. Pinto. Stochastic analyis of networked embedded systems. *United Technologies Research Center, Internal Report*, 2010.

[14] B. Plateau and K. Atif. Stochastic automata network of modeling parallel systems. 17(10):1093–1108, October 1991.

[15] J. Rank (Editor). *The estimation of copulas: Theory and practice, In. Copulas: From Theory to Application in Finance*. Risk Books, New York, 2007.

[16] C. Romano. *Calibrating and Simulating Copula Functions: An Application to the Italian Stock Market*. Universit a degli Studi di Roma, La Sapienza, 2002. Centro Interdipartimentale sul Diritto e lEconomia dei Mercati, Working Paper, n. 12.

[17] A. Sklar. Fonctions de repartition a n dimensions e leurs marges. *Publications de l'Institut de Statistique de l'Univiversite de Paris*, 8:229–231, 1959.

[18] M. Telek. *Some Advanced Reliability Modelling Techniques*. Ph.D. Thesis. Technical University of Budapest, Budapest, Hungary, 1994.

[19] P. Trivedi and D. M. Zimmer. Copula modeling: An introduction for practitioners. *Foundations and Trends in Econometrics*, 1(1):1111, 2005.

[20] C. Murray Woodside and Yao Li. Complete decomposition of stochastic petri nets representing generalized service networks. *IEEE Trans. Comput.*, 44(4):577–592, 1995.