A logic for model-checking mean-field models

Anna Kolesnichenko, Pieter-Tjerk de Boer, Anne Remke, and Boudewijn R. Haverkort

Centre for Telematics & Information Technology, University of Twente, Enschede, The Netherlands
{a.v.kolesnichenko, p.t.deboer, a.k.i.remke, b.r.h.m.haverkort}@utwente.nl

***Abstract*—Recently the mean-field method has been adopted for analysing systems consisting of a large number of interacting objects in computer science, biology, chemistry, etc. It allows for a quick and accurate analysis of such systems, while avoiding the state-space explosion problem. So far, the method has primarily been used for performance evaluation. In this paper, we use the mean-field method for model-checking. We define and motivate a logic *MF-CSL* for describing properties of systems composed of many identical interacting objects. The proposed logic allows describing both properties of the overall system and of a random individual object. Algorithms to check the satisfaction relation for all MF-CSL operators are proposed. Furthermore, we explain how the set of all time instances that fulfill a given MF-CSL formula for a certain distribution of objects can be computed.**

I. Introduction

In this paper, we introduce a logic and algorithms for model-checking mean-field models. This allows very efficient model-checking of very large systems, if they consist of many similar interacting objects.

The mean-field method ([1], [2], [3]) works by not considering the state of each individual object separately, but only their average, i.e., what fraction of the objects are in each possible state at any time. It allows to compute the exact limiting behavior of an infinite population of identical objects, and this exact limiting behavior is a good approximation when the number of objects is not infinite but sufficiently large. It can also be used to calculate the steady-state solution for some models. Examples of systems for which the mean-field method has been successfully applied include gossiping protocols [4], disease spread between islands [5], peer-to-peer botnets spread [6], the growth dynamics of individual vertices in scale-free random networks [7], etc.

Thus far, the mean-field method was used for performance evaluation of systems, but in this paper we want to explore the use of mean-field methods for *model-checking* such systems. Model-checking means checking whether a system state satisfies certain properties. It was initially introduced for finite deterministic models, for validation of computer and communication systems, and later extended towards stochastic models and models with continuous time.

Model-checking of large systems is made difficult by the state-space explosion problem. Since the mean-field method avoids this problem, it would be helpful if it could also be used for model-checking. One challenge for model-checking of mean-field models is the fact that they have a continuous state-space. Another challenge lies in the fact that the underlying Markov chain of a random individual is time-*in*homogeneous.

Addressing the above challenges, there is earlier work on model-checking models with continuous state variables, for example in [8], [9], [10], [11]. To the best knowledge of the authors model-checking of time-inhomogeneous CTMCs was not fully addressed, however, a number of algorithms limited to special cases were recently proposed. In [12] model-checking algorithms for the Hennessy-Milner Logic (HML) on time-inhomogeneous CTMC were proposed under the assumption of piecewise constant rates. In [13] model-checking LTL properties is addressed. Verification of time-bounded CSL properties of an individual object within a mean-field model was recently discussed in [14].

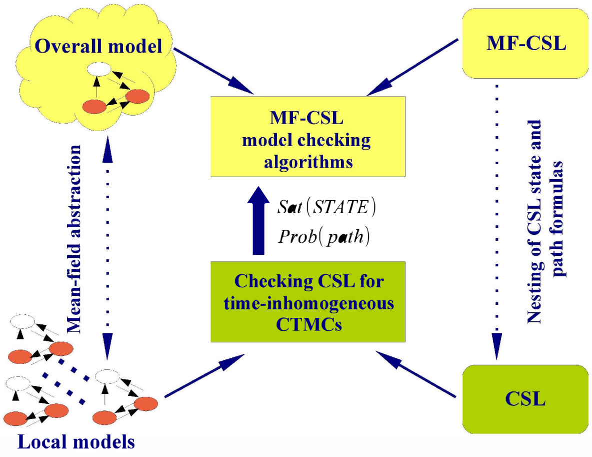
Overall model
MF-CSL
MF-CSL model checking algorithms
Sat(STATE)
Prob(path)
Checking CSL for time-inhomogeneous CTMCs
Mean-field abstraction
Nesting of CSL state and path formulas
CSL
Local models

Fig. 1. Overview of model-checking MF-CSL.

The main contribution of this paper is the introduction of the *Mean Field Continuous Stochastic Logic* (MF-CSL), and algorithms for checking MF-CSL properties. An MF-CSL formula describes a property of the *overall* model, in terms of what fraction of the individual objects satisfy a CSL formula on the *local* model. Thus, checking an MF-CSL formula requires first checking CSL formulas on the time-inhomogeneous CTMC describing the local model. This is illustrated in Figure 1.

The paper is further organized as follows: in Section II the a brief overview of the mean-field model definition and mean-field analysis is provided. The MF-CSL logic is introduced in Section III. Sections IV and V provide algorithms for model-checking local CSL formulas and global MF-CSL formulas respectively. Examples are given in Section VI. Conclusions are provided in Section VII.

978-1-4799-0181-4/13/$31.00 ©2013 IEEE

II. Mean-field approximation

The main idea of mean-field analysis is to describe the overall behavior of a system that is composed of many similar objects, via the average behavior of the individual objects. In Section II-A we define the *local model*, which describes the behavior of each individual object, and the way to build the mean-field *overall model* that describes the complete system. In Section II-B we recall how to compute transient and steady-state occupancy measures using mean-field analysis.

A. Model definition

The individual or local model is defined as follows:

Definition 1 (Local model) *A local model $\mathcal{M}^l$, describing the behavior of one object is constructed as a tuple $(S^l, \mathbf{Q}, L)$ that consists of a finite set of K local states $S^l = \{s_1, s_2, ..., s_K\}$; the infinitesimal generator matrix $\mathbf{Q}$ which may depend on the overall system state: $S^l \times S^l \times S^o \to \mathbb{R}$ (where S^o is the state space of the global model, to be introduced below); and the labeling function $L : S^l \to 2^{LAP}$ that assigns local atomic propositions from a fixed finite set LAP (Local Atomic Properties) to each state.* □

Note that self-loops are eliminated. Note also that the generator matrix $\mathbf{Q}$ is a two-dimensional matrix $S^l \times S^l$ and S^o is included in the definition in order to show the dependence on the overall system state.

The states of the local model denote the "modes" an individual component goes through during its lifetime; the transitions denote the changing between these states.

Once the local model is built, we use the mean-field method to model and analyze the overall behavior of N such objects. Instead of modeling each object individually, which would lead to the state-space explosion problem, we lump the state space and create *the overall mean-field model* $\mathcal{M}^O$ from the local model $\mathcal{M}^l$, as follows:

Definition 2 (Overall mean-field model) *An overall mean-field model $\mathcal{M}^O$ describes the behavior of $N \to \infty$ identical objects, each modeled by $\mathcal{M}^l$, and is defined as a tuple $(S^o, \mathbf{Q})$, that consists of an infinite set of states*

$$S^o = \{\overline{m} = (m_1, m_2, \ldots, m_K) | (\forall j \in \{1, \ldots, K\}, \\ (m_j \in [0,1]) \wedge (\sum_{j=1}^{K} m_j = 1))\},$$

where $\overline{m}$ is called occupancy vector; m_j denotes the fraction of the N individual objects that are in state s_j of the local model $\mathcal{M}^l$. The transition rate matrix $\mathbf{Q}(\overline{m})$ consists of entries $\mathbf{Q}_{s,s'}(\overline{m})$ that now describe the transition of fractions of objects from state s to state s'. □

When the number of interacting objects is finite but large, the mean-field model is an approximation of the real system behaviour. Note that for any finite N the occupancy vector $\overline{m}$ is a discrete distribution over K states, taking values in $\{0, \frac{1}{N}, \frac{2}{N}, \ldots, 1\}$, while for infinite N, the m_j are real numbers in $[0, 1]$. The matrix $\mathbf{Q}(\overline{m})$ of the overall model is the same as $\mathbf{Q}(\overline{m})$ of the local model $\mathcal{M}^l$.

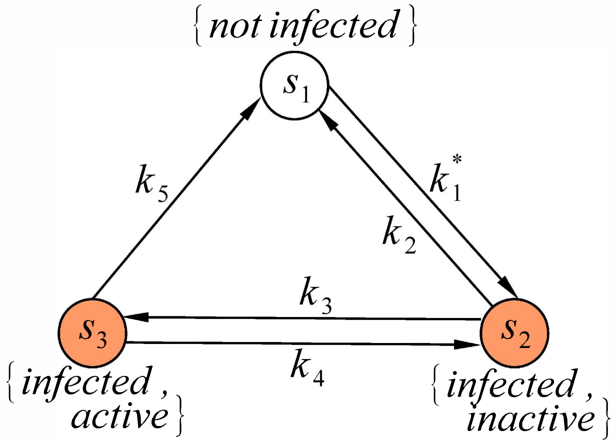

Fig. 2. Example of the CTMC describing computer virus spread.

To illustrate the relation between the local and overall mean-field model we address the following example, which will be used as running example throughout the paper.

Example 1 *Figure 2 shows a simplified version of the models used in [6], which describes the spread of a computer virus. States represent the modes of an individual computer, which can be* not-infected, infected and active *or* infected and inactive. *An infected computer is active when it is spreading the virus and inactive when it is not. This results in the finite local state space $S^l = \{s_1, s_2, s_3\}$ with $|S^l| = K = 3$ states. They are labelled as* infected, not infected, active *and* inactive, *as indicated in Figure 2.*

The transition rates k_1^*, k_2, k_3, k_4, k_5 represent the following: the infection rate k_1^*, the recovery rate for an inactive infected computer k_2, the recovery rate for an active infected computer k_5, and the rates with which computers become active k_3 and return to the inactive state k_4. Rates k_2, k_3, k_4, and k_5 are specified by the individual computer and computer virus properties and do not depend on the overall system state. The infection rate k_1^* does depend on the rate of attack (k_1), the fraction of computers that is infected and active and, possibly, the fraction of not-infected computers. The dependence on the overall system state should reflect the real-life scenario and might be different for different situations. We give two examples of how the infection rate can depend on the overall system state in typical cases. The infection rate might be seen as the number of attacks performed by all active infected computers, which is distributed over all not-infected computes, that is, $k_1^*(t) = k_1 \cdot \frac{m_3(t)}{m_1(t)}$. Note that we assume here that the computer viruses are "smart enough" to only attack computers which are not yet infected, see [15], [6]. In contrast, epidemiological models usually assume the infection rate to be proportional to the fraction of the actively infected computers and not to depend on the number of not-infected computers, leading to $k_1^*(t) = k_1 \cdot m_3(t)$.

Given a system of N such computers, we can model the average behavior of the whole system through the global mean-field model, which has the same underlying structure *as the individual model (see Figure 2), however, with state space* $S^o = \{m_1, m_2, m_3\}$*, where* m_1 *denotes the fraction of not-infected computers, and* m_2 *and* m_3 *denote the fraction of active and inactive infected computers, respectively. For example, a system without infected computers is in state* $\overline{m} = (1, 0, 0)$*; a system with 50% not infected computers and 40% and 10% of inactive and active computers, respectively, is in state* $\overline{m} = (0.5, 0.4, 0.1)$*. Note that* S^o *is a simplex in a three-dimensional space.*

B. Mean-field analysis

Given the local model $\mathcal{M}^l$ that represents the individual object and the overall model $\mathcal{M}^O$ for N identical objects, the behavior of the overall system is approximated using the mean-field method. In the following we briefly recall the main ideas of the mean-field approach; for more information see [6], [2], [1]. The mean-field approximation is based on Theorem 1 from [16], which states the following:

Theorem 1 (Mean-field convergence theorem) *The normalized occupancy vector* $\overline{m}(t)$ *at time* t *tends to be deterministic in distribution and satisfies the following differential equations when* N *tends to infinity:*

$$\frac{d\overline{m}(t)}{dt} = \overline{m}(t) \cdot \mathbf{Q}(\overline{m}(t)), \text{ given } \overline{m}(0). \tag{1}$$

□

The transient analysis of the overall system behavior can be obtained using the above system of differential equations (1), i.e., the fraction of the objects in each state of $\mathcal{M}^l$ at every time t is calculated, starting from some given initial occupancy vector $\overline{m}(0)$. Given the mean-field convergence results, it is easy to see that the "limit" local model $\mathcal{M}^l$ of a random object within the overall system is a time-*in*homogeneous continuous time Markov chain, whose rates depend on the overall system state $\overline{m}(t)$ via ODEs (1).

The stationary behavior of the overall system can in some cases be approximated using stationary points of the deterministic *fluid limit*, i.e., the unique stationary point of the ODE (1) (for more information see [17]). The stationary distribution $\widetilde{m}$, if it exists, is the solution of:

$$\widetilde{m} \cdot \mathbf{Q}(\widetilde{m}) = 0, \text{with } \; \widetilde{m} = \lim_{t \to \infty} \overline{m}(t). \tag{2}$$

In this paper we consider continuous-time mean-field models; however, discrete-time models are also often used for specific applications. The discrete-time mean-field model is similar to the continuous model, however, the local model is a discrete-time Markov chain (DTMC). Note that all the results in the present paper can easily be adapted to discrete-time mean-field models. For more information we refer to [4].

While transient and steady-state analysis are very useful to analyze the system's behavior over time, they are not sufficient to analyze more involved properties, like reachability. In the following sections we explore how to obtain more information using mean-field models coupled with model-checking techniques. The aim of the paper is developing the basis for model-checking mean-field models, therefore we do not discuss the accuracy or applicability of the mean-field method and construction of the proper model as such; we assume instead that the proper model was constructed before applying model-checking and that the validity of the mean-field approximation has been ascertained.

III. Mean-Field Continuous Stochastic Logic

Mean-field models include two layers of interest, namely, (i) the overall model itself, which describes the behavior of the system in terms of the fraction of local objects in a given state; and (ii) the local model, which describes the behavior of a local object. Recall that the local object in the mean-field model is modeled as a time-inhomogeneous CTMC, hence, Continuous Stochastic Logic (CSL) can be used to specify properties of interest of the local model $\mathcal{M}^l$. Recall the definition of CSL [18]:

Definition 3 ***Syntax of CSL.*** *Let* $p \in [0, 1]$ *be a real number,* $\bowtie \in \{\leq, <, >, \geq\}$ *a comparison operator,* $I \subseteq \mathbb{R}_{\geq 0}$ *a non-empty time interval and* LAP *a set of atomic propositions with* $lap \in LAP$*. CSL* ***state formulas*** Φ *are defined by:*

$$\Phi ::= tt \mid lap \mid \neg\Phi \mid \Phi_1 \wedge \Phi_2 \mid \mathcal{S}_{\bowtie p}(\Phi) \mid \mathcal{P}_{\bowtie p}(\phi),$$

where ϕ *is a* ***path formula*** *defined as:*

$$\phi ::= \chi^I \Phi \mid \Phi_1 \; U^I \; \Phi_2.$$

□

To define the semantics of path formulas we first recall the notion of a path as in [18]. An *infinite path*[1] σ is a sequence $s_0 \xrightarrow{t_0} s_1 \xrightarrow{t_1} s_2 \xrightarrow{t_2} \ldots$ with, for $i \in \mathbb{N}$; $s_i \in S^l$ and $t_i \in \mathbb{R}_{>0}$ such that $\mathbf{Q}_{(s_i, s_{i+1})}(\overline{m}(\sum_{j=0}^{i} t_j)) > 0$ for all i. A finite path σ is a sequence $s_0 \xrightarrow{t_0} s_1 \xrightarrow{t_1} \ldots s_{h-1} \xrightarrow{t_{h-1}} s_h$ such that s_h is absorbing, and $\mathbf{Q}_{(s_i, s_{i+1})}(\overline{m}(\sum_{j=0}^{i} t_j)) > 0$ for all $i < h$. For an infinite path σ, $\sigma[i] = s_i$ denotes for $i \in \mathbb{N}$ the $(i{+}1)st$ state of path σ. The time spent in state s_i is denoted by $\delta(\sigma; i) = t_i$. Moreover, with i the smallest index with $t \leq \sum_{j=0}^{i} t_j$, let $\sigma@t = \sigma[i]$ be the state occupied at time t. For finite paths σ with length $h + 1$, $\sigma[i]$ and $\delta(\sigma; i)$ are defined in the way described above for $i < h$ only and $\delta(\sigma; h) = \infty$ and $\delta@t = s_h$ for $t > \sum_{j=0}^{h-1} t_j$. $Path^{\mathcal{M}^l}(s_i, \overline{m})$ is the set of all finite and infinite paths of the CTMC that start in state s_i given the state $\overline{m}$ of the overall model $\mathcal{M}^l$ and $Path^{\mathcal{M}^l}(\overline{m})$ includes all (finite and infinite) paths of the CTMC, which depends on the overall system state (global time) if the CTMC is time-inhomogeneous. A probability measure $Pr(\overline{m})$ on paths can be defined as in [18].

When the local CTMC is time-homogeneous the semantics of CSL formulas is well known. However, in any non-trivial mean-field model, the transition rates of the local CTMC $\mathcal{M}^l$ are not constant. According to Definition 1 the rates of the local model $\mathcal{M}^l$ may depend on the state of the global model $\mathcal{M}^O$, which changes with time. There are two ways to formalize this: the local rates depend on (i) the *current state* $\overline{m}$, which changes with time, or (ii) on the *global* time. While the first is more intuitive, it does not allow transition rates to depend explicitly on global time. For ease of notation, in the

[1] Note that $\overline{m}(\sum_{j=0}^{i} t_j)$ is the global state of the overall model $\mathcal{M}^O$ at the time of the i'th transition.

following we restrict ourselves to models that only depend on the overall state. Note that our approach can easily be extended to models that explicitly depend on global time and the proposed algorithms can handle both cases.

Since the local model changes with the overall system state, the satisfaction relation for a local state or path depends on the global state $\overline{m}$, as follows:

Definition 4 ***Semantics of CSL.*** *Satisfaction of state and path CSL formulas for time-inhomogeneous CTMCs is given as follows:*

$$\begin{array}{ll}
s \models^{\overline{m}} tt & \forall s \in S^l, \\
s \models^{\overline{m}} lap & \text{iff } lap \in L(s), \\
s \models^{\overline{m}} \neg\Phi & \text{iff } s \not\models^{\overline{m}} \Phi, \\
s \models^{\overline{m}} \Phi_1 \wedge \Phi_2 & \text{iff } s \models^{\overline{m}} \Phi_1 \textit{ and } \; s \models^{\overline{m}} \Phi_2, \\
s \models^{\overline{m}} \mathcal{S}_{\bowtie p}(\Phi) & \text{iff } \pi^{\mathcal{M}^l}(s, Sat(\Phi, \widetilde{m})) \bowtie p, \\
s \models^{\overline{m}} \mathcal{P}_{\bowtie p}(\phi) & \text{iff } Prob^{\mathcal{M}^l}(s, \phi, \overline{m}) \bowtie p, \\
\sigma \models^{\overline{m}} \chi^I \Phi & \text{iff } \sigma[1] \textit{ is defined, and} \\
 & \sigma[1] \models^{\overline{m}(\delta(\sigma,0))} \Phi \wedge \delta(\sigma, 0) \in I, \\
\sigma \models^{\overline{m}} \Phi_1 \; U^I \; \Phi_2 & \text{iff } \exists t' \in I : (\sigma@t' \models^{\overline{m}(t')} \Phi_2) \\
 & \wedge(\forall t'' \in [0, t')(\sigma@t'' \models^{\overline{m}(t'')} \Phi_1)),
\end{array}$$

where $\overline{m}$ is the occupancy vector (state of the overall model) at time 0, and $\overline{m}(t)$ is the occupancy vector at time t; $I \subseteq \mathbb{R}_{\geq 0}$ is a non-empty time interval and $Sat(\Phi, \overline{m}) = \{s' \in S^l : s' \models^{\overline{m}} \Phi\}$. $\pi^{\mathcal{M}^l}(s, Sat(\Phi, \widetilde{m})) = \sum_{s_j \in Sat(\Phi, \widetilde{m})} \pi^{\mathcal{M}^l}(s, s_j, \widetilde{m})$, describes the steady state probability to be in a state from $Sat(\Phi, \widetilde{m})$, where $\widetilde{m}$ is a stationary distribution.
$Prob^{\mathcal{M}^l}(s, \phi, \overline{m})$ is the probability measure of all paths $\sigma \in Path^{\mathcal{M}^l}(s, \overline{m})$ that satisfy ϕ when the system is starting in state s, that is, $Prob^{\mathcal{M}^l}(s, \phi, \overline{m}) = Pr\{\sigma \in Path^{\mathcal{M}^l}(s, \overline{m}) | \sigma \models^{\overline{m}} \phi\}$. □

Although $\overline{m}$ is referred to as the $\overline{m}$ vector at time 0, this is only for ease of discussion, without loss of generality. In fact, the $\overline{m}$ argument to $\models$ is just the global state at the time at which one checks the satisfaction relation. This is illustrated in the above definitions for χ and U, in which satisfaction at a future time t' is denoted by writing $\models^{\overline{m}(t')}$. Throughout the definition, $\overline{m}(t)$ is the occupancy vector at future time t, which can be obtained by solving the ODEs (1) for time t with $\overline{m}$ as initial condition.

Recall that since not in all models the mean-field approximation is valid for the steady-state; clearly, the steady-state operator should only be used for models in which it is.

The properties of interest of the mean-field model differ from the properties which can be described by CSL; therefore, in order to reason at the level of the overall model in terms of fractions of objects we introduce an extra layer "on top of CSL" that defines the logic *CSL for mean-field models*, which we call MF-CSL. The latter is able to describe the behaviour of the overall system in terms of the behaviour of random local objects.

Definition 5 ***Syntax of MF-CSL.*** *Let $p \in [0, 1]$ be a real number, and $\bowtie \in \{\leq, <, >, \geq\}$ a comparison operator. MF-CSL formulas Ψ are defined as follows:*

$$\Psi ::= tt \mid \neg\Psi \mid \Psi_1 \wedge \Psi_2 \mid \mathbb{E}_{\bowtie p}(\Phi) \mid \mathbb{E}\mathbb{S}_{\bowtie p}(\Phi) \mid \mathbb{E}\mathbb{P}_{\bowtie p}(\phi),$$

where Φ is a ***CSL state formula*** *and ϕ is a* ***CSL path formula****.*

□

We have introduced three expectation operators: $\mathbb{E}_{\bowtie p}(\Phi)$, $\mathbb{E}\mathbb{S}_{\bowtie p}(\Phi)$ and $\mathbb{E}\mathbb{P}_{\bowtie p}(\phi)$, with the following interpretation:

- $\mathbb{E}_{\bowtie p}(\Phi)$ denotes whether the fraction of objects that are in a (local) state satisfying a general CSL state formula Φ fulfills $\bowtie p$;
- $\mathbb{E}\mathbb{S}_{\bowtie p}(\Phi)$ denotes whether the fraction of objects that satisfy Φ in steady state, starting from the current distribution of objects, fulfills $\bowtie p$;
- $\mathbb{E}\mathbb{P}_{\bowtie p}(\phi)$ denotes whether the probability of taking a ϕ-path from a given distribution of objects over local states fulfills $\bowtie p$.

The interpretation of the probability operator $\mathbb{E}\mathbb{P}_{\bowtie p}(\phi)$ can be rephrased as the probability of a random object to satisfy path-formula ϕ. The formal definition of the MF-CSL semantics is as follows:

Definition 6 ***Semantics of MF-CSL.*** *The satisfaction relation $\models$ for MF-CSL formulas and states $\overline{m} = (m_1, m_2, \ldots, m_K) \in S^o$ of the overall mean-field model is defined by:*

$$\begin{array}{ll}
\overline{m} \models tt & \forall \;\; \overline{m} \;\; \in \;\; S^o, \\
\overline{m} \models \neg\Psi & \text{iff } \overline{m} \not\models \Psi, \\
\overline{m} \models \Psi_1 \wedge \Psi_2 & \text{iff } \overline{m} \models \Psi_1 \wedge \overline{m} \models \Psi_2, \\
\overline{m} \models \mathbb{E}_{\bowtie p}(\Phi) & \text{iff } \left(\sum\limits_{j=1}^{K} m_j \cdot Ind_{(s_j \models^{\overline{m}} \Phi)} \right) \bowtie p, \\
\overline{m} \models \mathbb{E}\mathbb{S}_{\bowtie p}(\Phi) & \text{iff } \left(\sum\limits_{j=1}^{K} m_j \cdot \pi^{\mathcal{M}^l}(s_j, Sat(\Phi, \overline{m})) \right) \bowtie p, \\
\overline{m} \models \mathbb{E}\mathbb{P}_{\bowtie p}(\phi) & \text{iff } \left(\sum\limits_{j=1}^{K} m_j \cdot Prob^{\mathcal{M}^l}(s_j, \phi, \overline{m}) \right) \bowtie p,
\end{array}$$

where $Sat(\Phi, \overline{m})$, $\pi^{\mathcal{M}}(s, Sat(\Phi, \overline{m}))$, $Prob^{\mathcal{M}^l}(s, \phi, \overline{m})$ are defined as in Definition 4; and $Ind_{(s_j \models^{\overline{m}} \Phi)}$ is an indicator function, which shows whether a local state $s_j \in S^l$ satisfies formula Φ for a given overall state $\overline{m}$:

$$Ind_{(s_j \models^{\overline{m}} \Phi)} = \left\{ \begin{array}{ll} 1, & \textit{if} \;\; s_j \models^{\overline{m}} \Phi, \\ 0, & \textit{if} \;\; s_j \not\models^{\overline{m}} \Phi. \end{array} \right.$$

□

To check an MF-CSL formula on the global level, the local CSL formula has to be checked first, and the results are then used on the global level. As discussed above, the local model $\mathcal{M}^l$ is a time-inhomogeneous CTMC, i.e., transition rates vary with the state of the overall model $\mathcal{M}^O$, which makes model-checking on the local level non-trivial. We discuss model-checking CSL formulas on the local model $\mathcal{M}^l$ in Section IV. The algorithms to compute the satisfaction of the MF-CSL formulas on the global model are presented in Section V.

Example 2 *To illustrate the expressive power of MF-CSL for mean-field models, consider the following MF-CSL formulas for the virus spreading model, as introduced in Example 1:*

1) *To define atomic propositions on the level of the mean-field model the operator $\mathbb{E}_{\bowtie p}(lap)$ can be used. If the system is considered* infected *if more than 80% of the computers are infected, this can be expressed as $\mathbb{E}_{\geq 0.8}(infected)$.*
2) *The property "the probability of a random computer to be infected in steady-state is higher than 10%" is expressed as follows: $\mathbb{ES}_{\geq 0.1}(infected)$. This property might be rephrased as "the fraction of computers, which are infected in steady state is at least 10%".*
3) *The property "the probability of an infected computer to recover (that is, change state from infected to not-infected) within five time units is less than 40%" is expressed as $\mathbb{EP}_{<0.4}(infected\ U^{[0;5]}\ not\text{-}infected\ )$.*

IV. Checking CSL formulae on the local level

In this section we first recall algorithms for model-checking CSL on time-homogeneous Markov chains in Section IV-A. The CSL operators which require a different approach for the time-inhomogeneous local model are discussed in Sections IV-B, IV-C, and IV-D. The satisfaction set development for a given CSL formula on a local model $\mathcal{M}^l$ is addressed in Section IV-E.

A. CSL for time-homogeneous CTMCs

All CSL operators can be divided into two groups:

- *time-independent operators*: lap[2], $\neg$, $\wedge$.
- *time-dependent operators*: $\mathcal{P}_{\bowtie p}$ and $\mathcal{S}_{\bowtie p}$, where $\mathcal{P}_{\bowtie p}$ includes path operators X and U.

The CSL operators can be nested according to Definition 3. Model-checking of the CSL formula is done by building the *parse tree* and performing the satisfaction set development of the individual operators recursively, as described in [18].

All time-independent CSL operators can be checked using the standard methods (see [18]) due to the independence of the results on time. Therefore, model-checking these operators is not included in the further discussion.

Properties that include the Next operator are rarely used in a real-life scenarios, therefore, we omit the discussion of such formulas and refer to [19] for algorithms for checking the CSL Next operator on the local time-inhomogeneous CTMC.

A discussion on the steady-state operator on the local mean-field model $\mathcal{M}^l$ is provided in Section IV-D.

Since the main challenge lies in model-checking the time-dependent operators, let us recall the interval until formula $\Phi_1 U^{[t_1,t_2]}\Phi_2$ for an arbitrary time-homogeneous CTMC $\mathcal{M}$, as in [18]. For model-checking such a CSL formula, we need to consider all possible paths, starting in a Φ_1 state at the current time and reaching a Φ_2 state during the time interval $[t_1, t_2]$ by only visiting Φ_1 states on the way. We can split such paths in two parts: the first part models the path from the starting state s to a Φ_1 state s_1 and the second part models the path from s_1 to a Φ_2 state s_2 only via Φ_1 states. We therefore need two transformed CTMCs[3]: $\mathcal{M}[\neg\Phi_1]$ and $\mathcal{M}[\neg\Phi_1 \vee \Phi_2]$, where $\mathcal{M}[\neg\Phi_1]$ is used in the first part of the path, for $t \in [0, t_1]$ and $\mathcal{M}[\neg\Phi_1 \vee \Phi_2]$ is used in the second, for $t \in [t_1, t_2]$. In the first part of the path, we only proceed along Φ_1 states, thus all states that do not satisfy Φ_1 do not need to be considered and can be made absorbing. As we want to reach a Φ_2 state via Φ_1 states in the second part, we can make all states that do not fulfill Φ_1 absorbing, because we are done as soon as we reach such a state.

In order to calculate the probability for such a path, we accumulate the multiplied transition probabilities for all triples (s, s_1, s_2), where $s_1 \models \Phi_1$ and is reached before time t_1 and $s_2 \models \Phi_2$ and is reached within time $t_2 - t_1$. Note that this can only be done for time-homogeneous CTMCs.

$$\text{Prob}^{\mathcal{M}}(s, \Phi_1 U^{[t_1,t_2]}\Phi_2) = \sum_{s_1 \models \Phi_1} \sum_{s_2 \models \Phi_2} \pi_{s,s_1}^{\mathcal{M}[\neg\Phi_1]}(t_1) \cdot \pi_{s_1,s_2}^{\mathcal{M}[\neg\Phi_1 \vee \Phi_2]}(t_2 - t_1). \quad (3)$$

Hence, CSL until formulas can be solved as a combination of two reachability problems, as shown in Equation (3), namely $\pi_{s,s_1}^{\mathcal{M}[\neg\Phi_1]}(t_1)$ and $\pi_{s_1,s_2}^{\mathcal{M}[\neg\Phi_1 \vee \Phi_2]}(t_2 - t_1)$ that can be computed by performing transient analysis on the transformed CTMCs. Note that Equation (3) is valid for $t_1 > 0$ and $t_2 > 0$, if $t_1 = 0$ the first reachability problem $\pi_{s,s_1}^{\mathcal{M}[\neg\Phi_1]}(t_1)$ is omitted.

B. Single until for time-inhomogeneous CTMCs

Due to the time-inhomogeneity of the local mean-field model, standard methods for model-checking timed operators can not be used. Recently, model-checking algorithms for a time bounded fragment of CSL were proposed in [14]. We adapt the model-checking algorithms presented in [14] for use on the local CTMC $\mathcal{M}^l$ of a mean-field model.

In the following we discuss model-checking of non-nested CSL interval until formulas on a time-inhomogeneous CTMC. This can occur in MF-CSL either being used in the expectation operator $\mathbb{E}_{\bowtie p}(P_{\bowtie q}(\Phi_1 U^{[t_1,t_2]}\Phi_2))$ or in the expectation probability operator $\mathbb{EP}_{\bowtie p}(\Phi_1 U^{[t_1,t_2]}\Phi_2)$ or expectation steady state operator $\mathbb{ES}_{\bowtie p}(P_{\bowtie q}(\Phi_1 U^{[t_1,t_2]}\Phi_2))$. Note that due to the restriction to single until formulas, the validity of Φ_1 and Φ_2 does not depend on time.

The core idea of CSL model-checking of until formulas as explained in the previous section remains unchanged for time-inhomogeneous CTMCs. However, due to time-inhomogeneity it is not enough to only consider the time duration, but the exact time at which the system is observed must be taken into account. Hence, we add time t' to the notation of a time-inhomogeneous reachability probability $\pi_{s,s_1}^{\mathcal{M}^l}(t', T)$ to denote that we start in state s at time t' and reach state s_1 within $T - t'$ time units.

[2] Note that the atomic property could be defined as a time-dependent operator, however according to Definition 2, it belongs to the time-independent group.

[3] We reuse the notation for a modified CTMC from [18], where the formula in the brackets refers to the set of states which are made absorbed

An arbitrary until formula $\Phi_1 U^{[t_1,t_2]}\Phi_2$ is then again solved by computing two reachability problems on the transformed local models $\mathcal{M}^l[\neg\Phi_1]$ and $\mathcal{M}^l[\neg\Phi_1 \vee \Phi_2]$, respectively:

$$\begin{aligned}&\text{Prob}^{\mathcal{M}^l}(s, \Phi_1 U^{[t_1,t_2]}\Phi_2, \overline{m}) = \\ &\sum_{s_1 \models^{\overline{m}} \Phi_1} \sum_{s_2 \models^{\overline{m}} \Phi_2} \pi_{s,s_1}^{\mathcal{M}^l[\neg\Phi_1]}(0, t_1) \cdot \pi_{s1,s2}^{\mathcal{M}^l[\neg\Phi_1 \vee \Phi_2]}(t_1, t_2 - t_1).\end{aligned} \tag{4}$$

Note that the first reachability probability always has zero as a starting time since we assume that $\overline{m}$ is the starting distribution[4].

In the following we describe how to compute the reachability probability $\pi_{s,s_1}^{\mathcal{M}^l}(t', T)$ for an arbitrary modified CTMC and a given occupancy vector $\overline{m}$ that is observed at time $t = 0$. Note that this can be used for both modified CTMCs $\mathcal{M}^l[\neg\Phi_1]$ or $\mathcal{M}^l[\neg\Phi_1 \vee \Phi_2]$, as needed in (4). We use Kolmogorov equations to perform the transient analysis on the modified CTMC, as described in [14].

Let $\Pi'(t', t' + T)$ be the probability matrix of the modified local CTMC, where $\Pi'_{s,s_1}(t', t' + T)$ is the probability of being in state s_1 at time $t' + T$, given that we were in state s at time t'. In order to find the transient probability the forward Kolmogorov equation is solved with the identity matrix as initial condition $\Pi(t', t' + 0)$:

$$\frac{d\Pi'(t', t' + T)}{d(T)} = \Pi'(t', t' + T) \cdot Q'(\overline{m}(t' + T)), \tag{5}$$

where $Q'(\overline{m}(t' + T))$ is the rate matrix of the modified CTMC.

Due to the modifications made in the local model, the transient probability matrix $\Pi'(t', t' + T)$ contains the reachability probabilities $\pi_{s,s_1}^{\mathcal{M}^l}(t', T)$ for all possible states s and s_1.

Once the reachability probabilities $\pi_{s,s_1}^{\mathcal{M}^l[\neg\Phi_1]}(0, t_1)$ and $\pi_{s_1,s_2}^{\mathcal{M}^l[\neg\Phi_1 \vee \Phi_2]}(t_1, t_2 - t_1)$ have been calculated using (5), the probability $\text{Prob}^{\mathcal{M}^l}(s, \Phi_1 U^{[t_1,t_2]}\Phi_2, \overline{m})$ can be computed according to Equation (4), which allows to check the satisfaction relation of a given occupancy vector $\overline{m}$ according to Definition 6.

Keeping the occupancy vector $\overline{m}$ and time t' as initial conditions of the mean-field model, the validity of a CSL formula may change when it is evaluated at a later moment in time $t \in [t', \theta]$, where θ is a predefined upper bound of the evaluation time. In the following we discuss how a reachability problem $\pi_{s,s_1}^{\mathcal{M}^l}(t, T)$ depends on its evaluation time t while T is kept constant.

First, the probability matrix $\Pi'(t', t' + T)$ is derived according to Equation (5), where t' is predefined. Next, the ODE describing the dependence of the transient probability on time t is derived by combining the forward and backward Kolmogorov equations using the chain rule:

$$\begin{aligned}\frac{d\Pi'(t, t + T)}{dt} &= -Q'(\overline{m}(t)) \cdot \Pi'(t, t + T) \\ &\quad + \Pi'(t, t + T) \cdot Q'(\overline{m}(t + T)).\end{aligned} \tag{6}$$

Finally, the time-dependent probability matrix $\Pi'(t, t + T)$ can be obtained by solving Equation (6) with initial condition $\Pi'(t', t' + T)$. This can be done either analytically or numerically, e.g., with the tool Wolfram Mathematica [20] as used in the current paper.

As mentioned above, the validity of a local CSL until formula may change when the system is evaluated at a later moment in time t due to the changing overall state. The time-dependent probability $\text{Prob}^{\mathcal{M}^l}(s, \Phi_1 U^{[t_1,t_2]}\Phi_2, \overline{m}, t)$ to take a $\Phi_1 U^{[t_1,t_2]}\Phi_2$ path in the local model $\mathcal{M}^l$, when starting in state s at time t, can be computed similar to Equation (4), by taking into account the time t that has elapsed since the initial condition $\overline{m}$ was observed:

$$\begin{aligned}&\text{Prob}^{\mathcal{M}^l}(s, \Phi_1 U^{[t_1,t_2]}\Phi_2, \overline{m}, t) = \\ &\sum_{s_1 \models^{\overline{m}} \Phi_1} \sum_{s_2 \models^{\overline{m}} \Phi_2} \pi_{s,s_1}^{\mathcal{M}^l[\neg\Phi_1]}(t, t + t_1) \cdot \\ &\pi_{s_1,s_2}^{\mathcal{M}^l[\neg\Phi_1 \vee \Phi_2]}(t + t_1, t + t_2 - t_1).\end{aligned} \tag{7}$$

Note that using Kolmogorov equations for solving reachability problems on the local models $\mathcal{M}^l$ is efficient due to the fact that the state space is usually quite small (see [14]).

C. Nested Until for time-inhomogeneous CTMCs

The method described in the previous section can be used when both Φ_1 and Φ_2 do not depend on time, i.e., when we do not have nested until formulas.

In the following let us consider the following nested until formula: $\mathcal{P}_{\bowtie p}(\Phi_1 U^{[t_\bullet, T]} \mathcal{P}_{\bowtie q}(\Phi_2 U^{[t_1,t_2]}\Phi_3))$. In order to solve a nested until formula the corresponding parse tree has to be built, as in the time-homogeneous case, and the satisfaction sets of all sub-formulas need to be computed. The satisfaction set of the sub-formula $\Gamma = \mathcal{P}_{\bowtie q}(\Phi_2 U^{[t_1,t_2]}\Phi_3)$, however, changes with time. To compute this set for a given $t \in [t_0, T]$, first $\text{Prob}^{\mathcal{M}^l}(s, \Phi_2 U^{[t_1,t_2]}\Phi_3, \overline{m}, t)$ needs to be computed for all $s \in \mathcal{S}^l$ according to Equation (7). Then the time-dependent satisfaction set of Γ is given by:

$$Sat(\Gamma, \overline{m}, t) = \{s \in \mathcal{S}^l \mid \text{Prob}^{\mathcal{M}^l}(s, \Gamma, \overline{m}, t) \bowtie p\}. \tag{8}$$

Having computed this set then in principle allows to model-check the nested until formula as a combination of two reachability problems, as in Equation (4). When replacing Φ_2 by Γ in this equation it becomes clear that model-checking a nested until formula requires computations on the modified CTMC $\mathcal{M}^l[\neg\Phi_1 \vee \Gamma]$. This is however not trivial, since the satisfaction set of Γ is time-dependent which results in a modified CTMC that also changes with time.

1) Time-varying set reachability: In the following, we describe how in general a time-bounded reachability problem $\pi^{\mathcal{M}[\neg\Gamma_1 \vee \Gamma_2]}(t', T)$ with time-dependent formulas Γ_1 and Γ_2 can be solved, similar to [14]. Note that t' indicates the starting

[4] For models which depend on global time this needs to be $\pi_{s,s_1}^{\mathcal{M}^l[\neg\Phi_1]}(t_0, t_1)$, where t_0 indicates that the system is observed at global time t_0.

time and T the duration of the time interval we are interested in.

At first we find the so-called discontinuity points, i.e., the time points $T_0 = t' \leq T_1 \leq T_2 \leq \cdots \leq T_k \leq T_{k+1} = T + t'$, where at least one of the satisfaction sets changes. Then we do the integration separately on each time interval $[T_i, T_{i+1}$ for $i = 0, ..., k$.

To ensure that only Γ_1 states are visited before a Γ_2 state is reached, we need to modify the CTMC $\mathcal{M}^l$ for each time interval as follows. First we introduce a new goal state s^*, which remains the same for all time intervals. Then, all Γ_1 and Γ_2 states are made absorbing and all transitions leading to Γ_2 states are readdressed to the new state s^*. Given this modified CTMC $\overline{\mathcal{M}^l}$, the transient probability matrix $\overline{\Pi'}(T_i, T_{i+1})$ is found for each time interval using the forward Kolmogorov equation, according to Equation (5).

Upon "jumps" between time intervals $[T_{i-1}, T_i]$ and $[T_i, T_{i+1}]$ it is possible that a state that satisfied Γ_1 in the previous time interval does not satisfy Γ_1 in the next. In this case the probability mass in this state is lost, since this path does not satisfy the reachability problem anymore. In the case that a state remains Γ_1 or a Γ_1 state is turned into a Γ_2 state the probability mass has to be carried over to the next time interval. This is described by the matrix $\zeta(T_i)$ of size $(|S^l| + 1) \times (|S^l| + 1)$ constructed in the following way: for each state $s \in S^l$ which satisfies $\neg\Gamma_1 \wedge \neg\Gamma_2$ before and after T_i it follows $\zeta(T_i)_{s,s} = 1$. For each state $s \in S^l$ which satisfies $\neg\Gamma_1 \wedge \neg\Gamma_2$ before T_i and Γ_2 after T_i we have $\zeta(T_i)_{s,s*} = 1$. For the new goal state s^* the entry always equals one ($\zeta(T_i)_{s*,s*} = 1$), and all other elements of $\zeta(T_i)$ are 0.

The probability to reach a Γ_2 state before time T has passed when starting in a $\neg\Gamma_2$ state at time t' is given then by the matrix $\Upsilon(t', t' + T)$:

$$\begin{aligned} \Upsilon(t', t' + T) =& \overline{\Pi'}(t', T_1) \cdot \zeta(T_1) \cdot \overline{\Pi'}(T_1, T_2) \cdot \\ & \zeta(T_1) \ldots \zeta(T_k) \cdot \overline{\Pi'}(T_k, t' + T). \end{aligned} \tag{9}$$

The probability to reach the goal state s^* is unconditioned on the starting state by adding 1 for all Γ_2 states:

$$\begin{aligned} \pi_{s,s^*}^{[\neg\Gamma_1 \vee \Gamma_2]}(t', t' + T) =& \Upsilon_{s,s^*}(t', t' + T) + \\ & \mathbb{1}\{s \in Sat(\Gamma_2, \overline{m}, t')\}. \end{aligned} \tag{10}$$

The way of calculating reachability probabilities as described above is based on the method proposed in [14]. The only difference is in the way of considering the probability mass which reaches the goal state. In the mentioned paper the state space is doubled and all goal states are considered separately, which increases the computational complexity and does not add any extra information. In our approach only one extra state is added in order to simplify the calculations.

Another way of reducing the computational complexity would be to lump all Γ_2 states and all $\neg\Gamma_1$ states in the model itself. However, in the case when the satisfaction sets of Γ_1 and Γ_2 change with time the state space of the modified local model will change at each discontinuity point, which would require more complicated calculation of (9) and (10).

2) Reachability probability as a function of time: To evaluate a nested until formula for varying points in time $t \in [t'; \Theta]$, in the following we adapt the two components of Equation (10) to allow for varying evaluation points.

Since only the first and the last component of $\Upsilon(t', t' + T)$ depend on t', we rewrite Equation (9) for ease of notation:

$$\Upsilon(t', t' + T) = \overline{\Pi'}(t', T_1) \cdot \Lambda(T_1, T_k) \cdot \overline{\Pi'}(T_k, t' + T), \tag{11}$$

where $\Lambda(T_1, T_k) = \zeta(T_1) \cdot \overline{\Pi'}(T_1, T_2) \cdot \cdots \cdot \overline{\Pi'}(T_{k-1}, T_k) \cdot \zeta(T_k)$.

To explicitly take into account the change of $\Upsilon(t, t + T)$ with time, the following differential equation is constructed using forward and backward Kolmogorov equations:

$$\frac{d\Upsilon(t, t + T)}{dt} = -\overline{Q}(t) \cdot \Upsilon(t, t + T) + \Upsilon(t, t + T) \cdot \overline{Q}(t + T), \tag{12}$$

where $\overline{Q}(t)$ is the rate matrix of $\overline{\mathcal{M}^l}$. Then in order to calculate $\Upsilon(t, t + T)$, the above is solved for $t \in [t', \theta]$. Note that when during the integration either t or $t + T$ reaches a discontinuity point T_i, the computation has to be stopped, $\Upsilon(t, t + T)$ has to be recomputed; the computation is resumed and ODE (12) is used until the next discontinuity point. The complete algorithm for this is given in the appendix. Note that the number of the discontinuity points is limited by the depth of nesting of the until-operator, which is low in practice, therefore the numerical complexity of the algorithm, described above is not an issue.

The time-dependent reachability probability can be computed as follows:

$$\begin{aligned} \pi_{s,s^*}^{[\neg\Gamma_1 \vee \Gamma_2]}(t, t + T) =& \Upsilon_{s,s^*}(t, t + T) + \\ & \mathbb{1}\{s \in Sat(\Gamma_2, \overline{m}, t)\}. \end{aligned} \tag{13}$$

Recall that the second component of this equation is also time-dependent and has to be reconsidered at each discontinuity point.

D. Steady-state operator

In the following we discuss how to model-check the steady state operator $\mathcal{S}_{\bowtie p}(\Phi)$ for a given overall distribution $\overline{m}$. Recall that this is only meaningful for mean-field models which are known to be also valid for the long run behaviour.

Since the long run behavior of the individual object reflects the behavior of the whole model, the stationary distribution $\widetilde{m}$ of the overall model can be used as the steady-state distribution of the local model $\pi^{\mathcal{M}^l}(s, s_j, \widetilde{m})$. Therefore, given the satisfaction set $Sat(\Phi, \widetilde{m})$ of the formula Φ, which can be found as will be explained in the next section, the steady state operator can be checked according to Definition 4:

$$\pi^{\mathcal{M}}(s, Sat(\Phi, \widetilde{m})) = \sum_{s_j \in Sat(\Phi, \widetilde{m})} \pi^{\mathcal{M}^l}(s, s_j, \widetilde{m}) = \sum_{s_j \in Sat(\Phi, \widetilde{m})} \widetilde{m}_j. \tag{14}$$

The steady-state probability does not depend on time, therefore, the satisfaction relation on the steady-state operator does not depend on time and the probability of the formula to hold remains constant at all times:

$$\pi^{\mathcal{M}}(s, Sat(\Phi, \widetilde{m}), t) = \sum_{s_j \in Sat(\Phi, \widetilde{m})} \widetilde{m}_j. \tag{15}$$

E. Satisfaction set development for the local model $\mathcal{M}^l$

The satisfaction set of a CSL formula on a time-inhomogeneous CTMC is constructed using a parse tree [18], as in the time-homogeneous case. First the satisfaction sets of the sub-formulas have to be developed. For time-independent operators nothing changes compared to [18], therefore we do not discuss this here. For time-dependent operators both the satisfaction set for a given time t' and the time-dependent satisfaction set for a given time interval $[t', \theta]$ can be computed, as follows.

For a given time t' and the overall system state $\overline{m}$ we obtain satisfaction set of the probability operator:

$$Sat(\mathcal{P}_{\bowtie p}(\phi), \overline{m}) = \{s \mid Prob^{\mathcal{M}^l}(s, \phi, \overline{m}) \bowtie p\}, \qquad (16)$$

where $Prob^{\mathcal{M}^l}(s, \phi, \overline{m})$ is given by Equation (4). According to Equation (14) the satisfaction set of the steady-state operator is as follows:

$$Sat(\mathcal{S}_{\bowtie p}(\Phi), \overline{m}) = \{s \mid \sum_{s_j \in Sat(\Phi, \overline{m})} \widetilde{m}_j \bowtie p\}. \qquad (17)$$

The time-dependent satisfaction set is developed similarly, but Equations (7) and (15) are used for the probability and steady-state operator respectively:

$$Sat(\mathcal{P}_{\bowtie p}(\phi), \overline{m}, t) = \{s \mid Prob^{\mathcal{M}^l}(s, \phi, \overline{m}, t) \bowtie p\}, \qquad (18)$$

$$Sat(\mathcal{S}_{\bowtie p}(\Phi), \overline{m}, t) = \{s \mid \sum_{s_j \in Sat(\Phi, \overline{m}, t)} \widetilde{m}_j \bowtie p\}. \qquad (19)$$

V. MF-CSL model-checking on the global level

Model-checking MF-CSL formula consists of two parts: checking the satisfaction relation for individual states and developing the *satisfaction set* of a given MF-CSL formula Ψ. Both parts include CSL model-checking on the local level, which is a non-trivial task if the local model is time-inhomogeneous, as has been discussed in Section IV. In this section we proceed with the satisfaction relation and show how to build the satisfaction set of MF-CSL operators on the overall model $\mathcal{M}^O$.

A. Checking the satisfaction relation for individual states

Given the results on the local level, checking individual states of the global model can be done by straight-forward application of Definition 6. We briefly discuss checking the satisfaction relation between a given occupancy vector $\overline{m}$ and expectation operators in the following.

For the expectation operator $\mathbb{E}_{\bowtie p}(\Phi)$ the satisfaction set of the local CSL formula is used in order to define the indicator function, which allows to check the following inequality:

$$\left(\sum_{j=1}^{K} m_j \cdot Ind_{(s_j \models^{\overline{m}} \Phi)} \right) \bowtie p.$$

For the expected probability operator $\mathbb{EP}_{\bowtie p}(\phi)$ we check

$$\left(\sum_{j=1}^{K} m_j \cdot Prob^{\mathcal{M}^l}(s_j, \phi, \overline{m}) \right) \bowtie p,$$

where the probability $\text{Prob}^{\mathcal{M}^l}(s, \Phi, \overline{m})$ is computed as described in Section IV.

Since the long run behavior of an individual object reflects the behavior of the whole model, checking the satisfaction of a steady-state MF-CSL formula $\mathbb{ES}_{\bowtie p}(\Phi, \overline{m})$ simplifies to the following expression:

$$\sum_{j=1}^{K} \pi^{\mathcal{M}^l}(s_j, Sat(\Phi, \overline{m}, t)) \cdot m_j = \pi^{\mathcal{M}^l}(s, Sat(\Phi, \overline{m}, t)).$$

Hence, the expected steady-state operator on the global level mirrors the steady state operator on the local level, when the steady-state exists (see [17]). Therefore the stationary distribution $\widetilde{m}$ of the global model is used as steady-state distribution of the local model and the expected steady-state operator is checked using Equation (14):

$$\left(\sum_{s_j \in Sat(\Phi, \overline{m}, t)} \widetilde{m}_j \right) \bowtie p.$$

B. Satisfaction set development

Traditionally, the satisfaction set of a given formula is the set of states of the model which satisfies a given formula. In the context of MF-CSL model-checking, this would result in a set of all occupancy vectors $\overline{m}$ satisfying a given MF-CSL formula. While such a set can be built for time-independent MF-CSL operators, it is not a trivial task for time-dependent operators, since the model-checking on the local model $\mathcal{M}^l$ would have to be done without knowing the initial conditions, i.e., the occupancy vector. Theoretically speaking, in some cases the general solution of the ODEs (1) can be used, however, in practice these solutions are not easy (or even impossible) to find. Furthermore, the procedure of model-checking time-dependent CSL operators often includes numerical evaluation, therefore, using the general solution seems not feasible.

However, once the initial occupancy vector is fixed, the time instances where a MF-CSL formula holds when evaluated at later times $t \in [0, \theta]$ can be found. From this point of view, the *conditional satisfaction set* of the MF-CSL formula for a given occupancy vector $\overline{m}$ and time interval $[0, \theta]$ is defined as:

$$cSat(\Psi, \overline{m}, \theta) = \{t \in [0, \theta] \mid \overline{m}(t) \models \Psi\}. \qquad (20)$$

In the following, we discuss how to develop the conditional satisfaction set of MF-CSL expectation operator. Table I summarizes the equations which define the satisfaction set of the expectation operators. The algorithms for calculating satisfaction sets are given in the following.

A set of inequalities defines the constraints on the satisfaction set of the expectation operator $cSat(\mathbb{E}_{\bowtie p}\Phi, \overline{m}, \theta)$. To construct these inequalities one has to find the satisfaction set $Sat(\Phi, \overline{m}, t)$ of the local CSL state formula Φ (see Section IV-E).

Since $Sat(\Phi, \overline{m}, t)$ changes with time, the calculation is done piecewise, taking into account the discontinuity points $0 < \tau_1 < \tau_2 < \cdots < \tau_h < \theta$, where at least one state of the local model leaves or enters the satisfaction set.

MF-CSL operator	Set of inequalities to build $cSat(\Psi,\overline{m},\theta)$	Requires computation on $\mathcal{M}^l$
$\Psi = \mathbb{E}_{\bowtie p}(\Phi)$	$\begin{cases} \left[\sum_{j=1}^{K} m_j(t)\cdot Ind^{[\tau_i;\tau_{i+1}]}_{s_j\models^{\overline{m}}\Phi}(\overline{m},t)\right] \bowtie p, \\ \frac{d\overline{m}(t)}{dt} = \overline{m}(t)\mathbf{Q}(\overline{m}(t)), \\ \forall \tau_i \in [0,\Theta] \end{cases}$	$Sat(\Phi,\overline{m},t)$
$\Psi = \mathbb{ES}_{\bowtie p}(\Phi)$	$\begin{cases} \left[\sum_{s_j\in Sat(\Phi,\widetilde{m},t)} m_j(t)\cdot \widetilde{m}_j\right] \bowtie p, \\ \frac{d\overline{m}(t)}{dt} = \overline{m}(t)\mathbf{Q}(\overline{m}(t)), \end{cases}$	$Sat(\Phi,\widetilde{m},t)$
$\Psi = \mathbb{EP}_{\bowtie p}(\phi)$	$\begin{cases} \left[\sum_{j=1}^{K} m_j(t)\cdot Prob^{\mathcal{M}^l}(s_j,\phi,\overline{m},t)\right] \bowtie p, \\ \frac{d\overline{m}(t)}{dt} = \overline{m}(t)\mathbf{Q}(\overline{m}(t)), \end{cases}$	$Prob^{\mathcal{M}^l}(s_j,\phi,\overline{m},t)$

TABLE I. CONDITIONAL SATISFACTION SET DEVELOPMENT FOR THE MF-CSL FORMULAS

The indicator function $Ind^{[\tau_i;\tau_{i+1}]}_{s_j\models^{\overline{m}}\Phi}(\overline{m},t)$, which shows whether a local state s_j satisfies formula Φ, is then defined on each time interval $[\tau_i;\tau_{i+1}]$. The inequality is constructed at each time interval $[\tau_i;\tau_{i+1}]$ according to Definition 6.

The satisfaction set is developed as follows: for each time interval the constraints on the occupancy vector $\overline{m}(t)$ are found by solving the respective inequalities, where $t \in [\tau_i;\tau_{i+1}]$. Recall that ODEs (1) defines all possible occupancy vectors in the time interval $[\tau_i;\tau_{i+1}]$. These are checked against the above constraints and the time intervals at which the occupancy vector satisfies the inequalities are added to the satisfaction set. Note that in most cases only a numerical solution of ODEs (1) is available.

As mentioned in Section V-A, the steady-state operator on the global level mirrors the steady-state operator on the local level.[5] Therefore, for developing the satisfaction set of the global MF-CSL formula $\mathbb{ES}_{\bowtie p}(\Phi,\overline{m})$ the probability $\pi^{\mathcal{M}}(s, Sat(\Phi,\widetilde{m}),t)$, as found applying Equation (15), is used to build the respective inequalities. Note that this probability does not change with time. The overall distribution, given by the ODEs (1), is then checked against the above inequalities, and all the time instances where the inequalities are satisfied are added to the satisfaction set.

To construct the inequalities for $cSat(\mathbb{EP}_{\bowtie p}\phi,\overline{m},\theta)$ one has to find the probability measure $Prob^{\mathcal{M}^l}(s_j,\phi,\overline{m},t)$ of all paths that satisfy ϕ, when starting in state s_j (as given in Equation (7)). Note that in nearly all the cases $Prob^{\mathcal{M}}(s_j,\phi,\overline{m},t)$ can only be solved numerically. The inequalities describing $cSat(\mathbb{EP}_{\bowtie p}\phi,\overline{m},\theta)$ are constructed using Definitions 5 and 6.

The remaining MF-CSL operators can be checked as follows:

- $\Psi = tt$ then $cSat(\Psi,\overline{m},\theta) = [0,\theta]$;
- $\Psi = \Psi_1 \wedge \Psi_2$ then $cSat(\Psi,\overline{m},\theta) = cSat(\Psi_1,\overline{m},\theta) \cap cSat(\Psi_2,\overline{m},\theta)$;
- $\Psi = \neg\Psi_1$ then $cSat(\Psi\ ,\overline{m},\theta) = [0,\theta] \setminus cSat(\Psi_1,\overline{m},\theta)$.

By nesting formulas more complex measures of interest can be specified. Model-checking of nested MF-CSL formulas does not differ from CSL model-checking, and the parse tree of the MF-CSL formula Ψ is built as for CSL formulas and the model-checking procedure is invoked recursively.

VI. EXAMPLES

In this section some examples of checking MF-CSL formulas against a given occupancy vector $\overline{m}$ and finding the satisfaction set of MF-CSL formula are described. We use the model given in Example 1. Before starting the calculations we need to define the parameters of the model: the infection rate k_1^*, the recovery rate for inactive infected computers k_2, the recovery rate for active infected computer k_5, and the rates with which computers become active k_3 and return to the inactive state k_4 (see Figure 2). As was discussed in Section II-A, the rates of the local model may depend on the overall state of the system. In the following example the infection rate depends on the number of active infected computers, which spread the virus to the not-infected computers. The rate of infection for one computer is as follows:

$$k_1^*(t) = k_1 \cdot \frac{m_3(t)}{m_1(t)},$$

where $\overline{m}(t) = (m_1(t), m_2(t), m_3(t))$ represents the fraction of computers in each state, and k_1 is the attack rate of a single active infected computer.

Then Theorem 1 is used to derive the system of ODEs (1), that describes the mean-field model:

$$\begin{cases} \dot{m}_1(t) &=& -k_1 m_3(t) + k_2 m_2(t) + k_5 m_3(t), \\ \dot{m}_2(t) &=& (k_1 + k_4) m_3(t) - (k_2 + k_3) m_2(t), \\ \dot{m}_3(t) &=& k_3 m_2(t) - (k_4 + k_5) m_3(t). \end{cases} \tag{21}$$

The coefficients that are used in the following example are given in Setting 1 in Table II.

Let us consider the following formula

$$\Psi = \mathbb{EP}_{<0.3}(\text{not infected } U^{[0,1]} \text{ infected})$$

and a predefined occupancy vector $\overline{m} = (0.8, 0.15, 0.05)$.

In order to check the satisfaction relation $\overline{m} \models \Psi$ the following three steps are taken:

- using the ODEs (21) calculate the time-dependent rates of the local model $\mathcal{M}^l$;

[5] Recall that for time-inhomogeneous local CTMC the steady-state operator can only be used in a limited number of cases, because the stationary distribution of mean-field models can be approximated using stationary points of the ODEs (1) only if the model is well-behaved (for more information see e.g. [17]).

- perform the CSL model-checking on the local model $\mathcal{M}^l$ in order to compute $Prob^{\mathcal{M}^l}(s, \text{not infected} \quad U^{[0,1]} \quad \text{infected}, \overline{m})$ for all states $s \in S^l$;
- use Definition 6 to check the satisfaction relation $\overline{m} \models \Psi$.

The only time-dependent rate of the local model is $k_1^*(t) = k_1 \cdot \frac{m_3(t)}{m_1(t)}$, where $m_1(t)$ and $m_3(t)$ are the solution of the ODEs (21) with $\overline{m}$ as initial condition. Therefore the transition rate matrix $\mathbf{Q}(\overline{m}(t))$ equals

$$\mathbf{Q}(\overline{m}(t)) = \begin{pmatrix} -k_1 \cdot \frac{m_3(t)}{m_1(t)} & k_1 \cdot \frac{m_3(t)}{m_1(t)} & 0 \\ k_2 & -k_2 - k_3 & k_3 \\ k_5 & k_4 & -k_5 - k_4 \end{pmatrix}.$$

To find $Prob^{\mathcal{M}^l}(s, \text{not infected} \quad U^{[0,1]} \quad \text{infected}, \overline{m})$ the reachability problem $\pi_{s,s_1}^{\mathcal{M}^l[\neg\text{not infected}\vee\text{infected}]}(0,1) = \pi_{s,s_1}^{\mathcal{M}^l[\text{infected}]}(0,1)$ has to be solved according to the algorithm described in Section IV-B. The local model $\mathcal{M}^l$ is modified and all *infected* states are made absorbing. The Kolmogorov equation is used to calculate the transient probability matrix of the modified model, which consists of the reachability probabilities:

$$\Pi'(0,1) = \begin{pmatrix} 0.91 & 0.09 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

The probability of the until formula

$$\phi = \text{not infected } U^{[0,1]} \text{ infected}$$

to hold for each starting state is as follows: $Prob^{\mathcal{M}^l}(s_1, \phi, \overline{m}) = \pi_{s_1,s_2}^{\mathcal{M}^l[\text{infected}]}(0,1) + \pi_{s_1,s_3}^{\mathcal{M}^l[\text{infected}]}(0,1) = 0.09$; $Prob^{\mathcal{M}^l}(s_2, \phi, \overline{m})) = 0$; $Prob^{\mathcal{M}^l}(s_3, \phi, \overline{m})) = 0$.

According to Definition 6, the weighted sum of the entries of the occupancy vector $\overline{m}$ and the respective probabilities in the local model define the expected probability $\mathbb{E}\mathbb{P}(\phi)$: $\sum_{j=1}^{K} m_j \cdot Prob^{\mathcal{M}^l}(s_j, \phi, \overline{m}) = 0.8 \cdot 0.09 + 0.15 \cdot 0 + 0.05 \cdot 0 = = 0.072 < 0.3$. As one can see the occupancy vector $\overline{m} = (0.8, 0.15, 0.05)$ satisfies the formula $\mathbb{E}\mathbb{P}_{<0.3}(\text{not infected } U^{[0,1]} \text{ infected})$.

As was discussed in Section V, the satisfaction on the global MF-CSL formula may change with time. Let us consider the same formula $\Psi = \mathbb{E}\mathbb{P}_{<0.3}(\text{not infected } U^{[0,1]} \text{ infected})$ and occupancy vector $\overline{m} = (0.8, 0.15, 0.05)$. In the following we calculate the satisfaction set $cSat(\Psi, \overline{m}, 20)$ on the predefined time interval $[0, 20]$.

The calculation of the time-dependent probabilities $Prob^{\mathcal{M}^l}(s, \text{not infected } U^{[0,1]} \text{ infected}, \overline{m}, t)$ is done as described in Section IV-B. The model $\mathcal{M}^l$ is modified so the infected states are made absorbing. The transient probability $\Pi(0,1)$ is calculated as described above. Forward and backward Kolmogorov equations are used in order to construct the ODEs, describing the time-dependent transient probability of the modified model (see Equation (6)). These ODEs are solved using $\Pi(0,1)$ as initial condition. The solution of the ODEs defines the required reachability probabilities. The probabilities $Prob^{\mathcal{M}^l}(s, \text{not infected } U^{[0,1]} \text{ infected}, \overline{m}, t)$ are calculated using Equation (7). The time-dependent probability $Prob^{\mathcal{M}^l}(s_1, \text{not infected } U^{[0,1]} \text{ infected}, \overline{m}, t)$ is depicted in Figure 3. Starting at states s_2 and s_3 this probability equals zero at all times, since these states do not satisfy *not infected.*

Parameter		Setting 1	Setting 2
Attack	k_1	0.9	5
Inactive computer recovery	k_2	0.1	0.02
Inactive computers getting active	k_3	0.01	0.01
Active computer returns to inactive	k_4	0.3	0.5
Active computer recovery	k_5	0.3	0.5

TABLE II. PARAMETER SETTINGS.

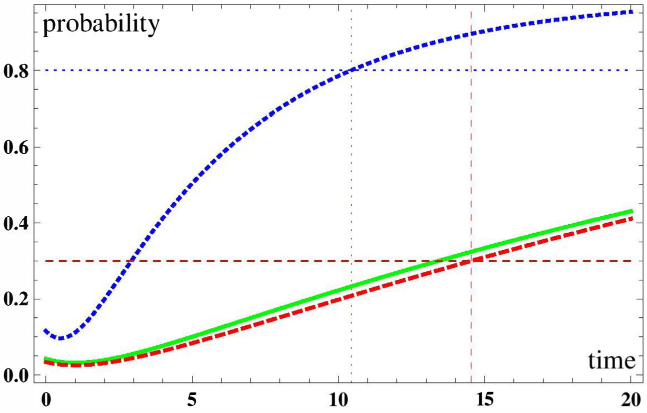

Fig. 3. The green solid line shows the probability at state s_1 $Prob^{\mathcal{M}^l}(s_1, \text{not infected} \quad U^{[0,1]} \quad \text{infected}, \overline{m}, t)$. The red dashed line shows the time-dependent expected probability for the formula $\mathbb{E}\mathbb{P}_{<0.3}(\text{not infected} \quad U^{[0,1]} \quad \text{infected})$. The time-dependent probability $Prob^{\mathcal{M}^l}(s_1, tt\ U^{[0,0.5]} \text{ infected}, \overline{m}, t)$ at state s_1 is presented by the blue dotted line.

To calculate the satisfaction set $cSat(\Psi, \overline{m}, 20)$ we construct the equation describing the dependence of the expected probability $\mathbb{E}\mathbb{P}(\text{not infected } U^{[0,1]} \text{ infected})$ on time: $\sum_{j=1}^{K} m_j \cdot Prob^{\mathcal{M}^l}(s_j, \phi, \overline{m}, t) = m_1(t) \cdot Prob^{\mathcal{M}^l}(s_1, \phi, \overline{m}, t) + m_2(t) \cdot 0 + m_3(t) \cdot 0$. The above probability is also depicted in Figure 3. As one can see, the formula $\mathbb{E}\mathbb{P}_{<0.3}(\text{not infected } U^{[0,1]} \text{ infected})$ holds for any time t between 0 and 14.5412, therefore, $Sat(\Psi, \overline{m}) = [0, 14.5412)$.

In the following we discuss a more involved example, which describes the "good" behavior of the system from the point of view of the malware developers. The parameters of the model used in this example are given in Setting 2 in Table II. We check the following satisfaction relation:

$$\begin{aligned}(0.85; 0.1; 0.05) \models \mathbb{E}_{>0.8}(\mathcal{P}_{>0.9}(\text{infected } U^{[0,15]} \\ (\mathcal{P}_{>0.8}\ tt\ U^{[0,0.5]} \text{ infected})))\wedge \mathbb{E}_{<0.1} \text{ active}.\end{aligned}$$

The parse tree of this formula has to be constructed and the sub-formulas $\Psi_1 =$

$\mathbb{E}_{\bowtie p}(\mathcal{P}_{>0.9}(\text{infected} \quad U^{[0,15]}(\mathcal{P}_{>0.8} tt \quad U^{[0,0.5]} \text{ infected})))$ and $\Psi_2 = \mathbb{E}_{<0.1}(\text{infected})$ have to be checked.

To model-check Ψ_1 we first check the CSL formula $\Phi = \mathcal{P}_{>0.9}(\text{infected } U^{[0,15]}(\mathcal{P}_{>0.8} tt \ U^{[0,0.5]} \text{ infected}))$ on the local level. And again, the formula is split and the time-dependent satisfaction set of the sub-formula $\Phi_1 = (\mathcal{P}_{>0.8} tt \ U^{[0,0.5]} \text{ infected})$ is calculated.

Similarly to the previous example, the probability $Prob^{\mathcal{M}^l}(s, tt \ U^{[0,0.5]} \text{ infected}, \overline{m}, t)$ is calculated for all states $s \in S^o$. In Figure 3 this probability at state s_1 is depicted; the probabilities at states s_2 and s_3 equal to one, since these states are already *infected*. We see that the time-dependent satisfaction set is $Sat(\Phi_1, \overline{m}, t) = \{s_2, s_3\}$ for all $t \in [0, 10.443]$ and $Sat(\Phi_1, \overline{m}, t) = \{s_1, s_2, s_3\}$ for all $t \in (10.443, 15]$.

The next task is calculating the probability $Prob^{\mathcal{M}^l}(s, \text{infected} \quad U^{[0,15]}\Phi_1, \overline{m})$. The reachability probability for the time-varying satisfaction set of Φ_1 is calculated following the algorithm described in Section IV-C. We first calculate all discontinuity points $T_0 = 0$, $T_1 = 10.443$ and $T_2 = 15$. An extra state s^* is added and an indicator matrix $\zeta(T_1)$ is constructed: $\zeta(T_1)_{s^*,s^*} = 1$, $\zeta(T_1)_{s_1,s_2} = 0$ for all $s_1, s_2 \neq s^*$. The transient probabilities on time intervals $[0, 10.443)$ and $(10.443, 15]$ are calculated using forward Kolmogorov equation:

$$\Pi'(0, 10.443) = \begin{pmatrix} 0.53 & 0 & 0 & 0.47 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

$$\Pi'(10.443, 15 - 10.443) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Equation (9) is used to calculate $\Upsilon(0, 15)$:

$$\Upsilon(0, 15) = \begin{pmatrix} 0 & 0 & 0 & 0.47 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Equation (10) is used in order to calculate the reachability probability for each state $s \in S^o$: $\pi^{\mathcal{M}^l[\neg\text{infected}\vee\Phi_1]}_{s_1,s^*}(0, 15) = 0.47$; $\pi^{\mathcal{M}^l[\neg\text{infected}\vee\Phi_1]}_{s_2,s^*}(0, 15) = 1$; $\pi^{\mathcal{M}^l[\neg\text{infected}\vee\Phi_1]}_{s_3,s^*}(0, 15) = 1$. The probability $Prob^{\mathcal{M}^l}(s, \text{infected} \quad U^{[0,15]}\Phi_1, \overline{m})$ is calculated according to Equation (4), and equals to 0, 1, and 1 for states s_1, s_2, and s_3 respectively. Therefore the only states satisfying the formula $\mathcal{P}_{>0.9}(\text{infected } U^{[0,15]}\Phi_1)$ are s_2 and s_3.

The inequality for checking whether the given occupancy vector $\overline{m}$ satisfies the MF-CSL expectation formula $\mathbb{E}_{>0.8}(\mathcal{P}_{>0.9}(\text{infected } U^{[0,15]}(\mathcal{P}_{>0.8} \ tt \ U^{[0,0.5]} \text{ infected})))$ is then as follows: $0.85 \cdot 0 + 0.1 \cdot 1 + 0.05 \cdot 1 > 0.8$, which does not hold, therefore, $\overline{m} \not\models \Psi_1$.

As one can easily see, $\overline{m} \models \mathbb{E}_{<0.1}(\text{active})$ holds for $\overline{m} = (0.85; 0.1; 0.05)$, however, according to Definition 6, $\overline{m} \not\models \Psi_1 \wedge \Psi_2$, since $\overline{m} \not\models \Psi_1$.

VII. Conclusions

In this paper, we have introduced a logic and algorithms for doing model-checking of mean-field models.

The mean-field method allow efficient modeling of systems consisting of a large number of identical interacting components, by describing not each individual component, but their average. Since the details of the individual components are no longer visible in a mean-field description, existing logics are not suitable to express their properties. Therefore, we have introduced a new logic, called MF-CSL. This logic expresses properties of the *global* model, in terms of what fraction of objects satisfies *local* properties, where the latter are CSL-like properties of the individual objects.

Checking the local properties is challenging because the models of the individual objects are time-inhomogeneous Markov-Chains, as their parameters depend on the global state. We have adapted results from [14] to obtain algorithms for this, and building on this we have obtained algorithms for checking the global properties for a given global state, and for obtaining the time interval(s) in which a global property is satisfied.

A main limitation of the current algorithms is that they are not suitable for checking time-unbounded properties. Future work could aim to resolve this. However, the convergence of the mean-field model is not necessarily uniform in time, which means that for many models the mean-field approximation is not good in the limit of large time, in which case algorithms dealing with unbounded time are of course meaningless. A similar remark holds for steady-state properties; we have included algorithms for them in the current paper, but again these are only meaningful for models of which it is known (through means that are beyond the scope of this paper) that the mean-field approximation is valid for large time.

References

[1] T. Kurtz, "Solutions of Ordinary Differential Equations as Limits of Pure Jump Markov Processes," *Journal of Applied Probability*, vol. 7, no. 1, pp. 49–58, 1970.

[2] J. Le Boudec, D. McDonald, and J. Mundinger, "A generic mean field convergence result for systems of interacting objects," in *QEST*. IEEE CS Press, 2007, pp. 3–18.

[3] R. Darling, "Fluid limits of pure jump markov processes: a practical guide," Tech. Rep., 2002.

[4] R. Bakhshi, L. Cloth, W. Fokkink, and B. Haverkort, "Mean-Field Analysis for the Evaluation of Gossip Protocols," in *QEST*. IEEE CS Press, 2009, pp. 247–256.

[5] R. Bakhshi, J. Endrullis, S. Endrullis, W. Fokkink, and B. Haverkort, "Automating the mean-field method for large dynamic gossip networks," in *QEST*. IEEE CS Press, 2010, pp. 241–250.

[6] A. Kolesnichenko, A. Remke, P. de Boer, and B. Haverkort, "Comparison of the mean-field approach and simulation in a peer-to-peer botnet case study," in *EPEW*, ser. LNCS, vol. 6977. Springer, 2011, pp. 133–147.

[7] A. L. Barabási, R. Albert, and H. Jeong, "Mean-field theory for scale-free random networks," *Physica A: Statistical Mechanics and its Applications*, vol. 272, no. 1-2, pp. 173 – 187, 1999.

[8] P. Abdulla, K. Cerans, B. Jonsson, and Y. K. Tsay, "General decidability theorems for infinite-state systems," in *LICS*, 1996, pp. 313 –321.

[9] P. Wolper and B. Boigelot, "Verifying systems with infinite but regular state spaces," in *CAV*, ser. LNCS, vol. 1427. Springer, 1998, pp. 88–97.

[10] N. Halbwachs, Y. E. Proy, and P. Roumanoff, "Verification of Real-Time Systems using Linear Relation Analysis," *Formal Methods in System Design*.

[11] M. Gribaudo and A. Horváth, “Model checking functional and performability properties of stochastic fluid models,” *Electronic Notes in Theoretical Computer Science*, vol. 128, no. 6, pp. 295–310, 2005.

[12] J. Katoen and A. Mereacre, “Model Checking HML on Piecewise-Constant Inhomogeneous Markov Chains,” in *FORMATS*. Springer, 2008, pp. 203–217.

[13] T. Chen, T. Han, J. Katoen, and A. Mereacre, “LTL Model Checking of Time-Inhomogeneous Markov Chains,” in *ATVA*. Springer, 2009, pp. 104–119.

[14] L. Bortolussi and J. Hillston, “Fluid model checking,” in *CONCUR*, ser. LNCS, vol. 7454. Springer, 2012, pp. 333–347.

[15] E. van Ruitenbeek and W. H. Sanders, “Modeling peer-to-peer botnets,” in *QEST*. IEEE CS Press, 2008, pp. 307–316.

[16] A. Bobbio, M. Gribaudo, and M. Telek, “Analysis of Large Scale Interacting Systems by Mean Field Method,” in *QEST*. IEEE Computer Society, 2008, pp. 215–224.

[17] J. Le Boudec, “The Stationary Behaviour of Fluid Limits of Reversible Processes is Concentrated on Stationary Points,” Tech. Rep., 2010.

[18] C. Baier, B. Haverkort, H. Hermanns, and J. Katoen, “Model-checking algorithms for continuous-time Markov chains,” *IEEE Trans. Softw. Eng.*, vol. 29, no. 7, pp. 524–541, 2003.

[19] L. Bortolussi and J. Hillston, “Fluid model checking,” Tech. Rep., 2013.

[20] Wolfram Research, Inc., “Mathematica Tutorial,” http://www.wolfram.com/mathematica/, 2010.

APPENDIX

In the following the algorithm for calculation time-dependent reachability probabilities for a varying satisfaction sets of the sub-formulas is presented. First, the following three steps have to be taken to prepare the piece-wise integration:

1) All the discontinuity points $t' = T_0 < T_1 < \cdots < T_k < T_{k+1} = \theta + T$ are found. In addition the points $T'_i = T_i + T$ are considered for all $i = 0, 1, \ldots, k$ and $pre(T'_i)$ is defined as the largest T_j preceding T'_i and $post(T'_i)$ is defined as the smallest T_j after T'_i.
2) The probability matrices $\overline{\Pi}'(T_i, T_{i+1})$ and $\overline{\Pi}'(pre(T'_i), T'_i)$ are calculated for all $i \leq k$ using the forward Kolmogorov equation.
3) For each discontinuity point T_i, the matrix $\zeta(T_i)$ is computed as defined in Section IV-C, for all $i = 1, 2, .., k$.

When integrating Equation (12) for all $t \in [t', \theta]$, due to the discontinuity points, we may not have a single solution $\Upsilon(t, t+T)$ that can be used for all values of t. For the intervals between discontinuity points, $\Upsilon(t, t+T)$ is given by the solution of ODE (12). At each discontinuity point $\Upsilon(T_i, T_i + T)$ is recalculated and the integration is resumed until the next discontinuity point is reached.

At the first discontinuity point, i.e., $T_0 = t'$, $\Upsilon(t', t' + T)$ is given by Equation (9), and for all $T_0 \leq t \leq t^* = \min\{T_1, post(T'_0) - T\}$ $\Upsilon(t, t+T)$ is given by the solution of ODE (12). Then $\Upsilon(t^*, t^* + T)$ is recalculated, depending on whether t or $t + T$ hit a discontinuity point T_i.

- If $t^* = T_i$, then $\Upsilon(T_i; T'_i)$ has to be recomputed as follows:
$$\Upsilon(T_i, T'_i) = \overline{\Pi}'(T_i, T_{i+1}) \cdot \Lambda(T_{i+1}, pre(T'_i)) \cdot \overline{\Pi}'(pre(T'_i), T'_i).$$
The integration of the ODE (12) is resumed and $\Upsilon(t, t+T)$ is calculated for $T_i \leq t \leq \min\{T_{i+1}, post(T'_i) - T'_i + T_i\}$ until the next discontinuity point is reached.
- If $t^* + T = T_i$, then to account for the changes at the discontinuity point $\Upsilon(T_i - T; T_i)$ has to be multiplied on the right by $\zeta(T_i)$. The integration of the ODE (12) is resumed and $\Upsilon(t, t+T)$ is calculated for $T_i - T \leq t \leq \min\{post(T_i - T), T_{i+1} - T\}$ until the next discontinuity point is reached.

This procedure is repeated until the time bound of the evaluation $t = \theta$ is reached.