Rule-Based Modelling and Simulation of Drug-Administration Policies

Luca Bortolussi[*†], Thilo Krüger[*], Thorsten Lehr[**] and Verena Wolf[*]
[*]**Modelling and Simulation Group, Department of Computer Science, Saarland University, Germany**
[**]**Clinical Pharmacy, Department of Pharmacy, Saarland University, Germany**
[†]**DMG, University of Trieste, Italy**
{**luca@cs, thilo@dennis.cs, thorsten.lehr@mx, wolf@cs**}**.uni-saarland.de**

Keywords: Alzheimer's Disease, Monte-Carlo Simulation, Rejection-Based Simulation Algorithms, Rule-Based Modelling, Time-Dependent Rates

Abstract

We consider rule-based models extended with time-dependent reaction rates, a suitable formalism to describe the effect of drug administration on biochemical systems. In the paper, we provide a novel and efficient rejection-based simulation algorithm that samples exactly the trajectory space of such models. Furthermore, we investigate a model of drug administration in the context of a plaque-formation process of Alzheimer's disease, a polymerization process best described by a rule-based model to counteract the intrinsic combinatorial explosion of the underlying reaction network. Furthermore, time-dependent rates are needed to model the effect of drugs. We apply the simulation algorithm to study the efficacy of different drug administration policies.

1. INTRODUCTION

Stochastic modelling in systems biology is gaining momentum due to the recognition of the importance of noise-induced effects in biological systems [1, 2], which cannot be captured by deterministic models. More broadly, stochasticity plays a role in a larger class of (bio)chemical processes, like modification of polymers via crosslinking and degradation [3]. In this paper, we are in particular interested in the formation of plaques in Alzheimer's Disease, a polymerization process, and in analyzing the effect of drugs that can potentially block their formation. This problem led us to a new and efficient simulation algorithm for rule-based modelling capable of treating drug-administration policies.

Stochastic modelling and simulation: Due to the inherent complexity of such stochastic models, reflected in their state space explosion, simulation is the most widespread analysis tool. A chemical reaction network is described as a *Continuous Time Markov Chain* (CTMC) and *Monte-Carlo* (MC) methods are used to generate sample traces, like the well known Direct Method, also called Gillespie's Algorithm [4, 5]. This approach can be implemented easily for simple chemical networks and it is proven to produce exact traces of the underlying CTMC. However, computational efficiency remains a central issue while simulating stochastic models. The first reason is the large number of simulated events due to fast reactions or due to a high number of reacting molecules. By using the Direct Method, after each event all reaction rates have to be updated and therefore high effort is needed to simulate models with many events. To combat this problem several approaches were developed, for example the exact *Next Reaction Method* [6] and the approximate τ-*Leaping* [7]. The former speeds up the computations that have to be done in each step, while the latter lumps events together and thus reduces the number of simulation steps.

Rule-based modelling: Another critical issue when simulating stochastic models in systems biology is the network size. All approaches described above require an explicit description of all individual biochemically active species and biochemical reactions [4, 5]. However, even simple systems can lead to a combinatorial explosion of species and reactions, and therefore simulation of such models with the Direct Method becomes infeasible. Well-known examples are large models of biochemical signaling processes [8, 9] and models of biochemical polymerizations [10]. Rule-based modelling is a viable alternative in this case. In this method, different but structurally similar reactions are combined into rules [8, 9, 11]. For example, an activation of a functional group can be possible with the same rate independent of the chemical environment of the functional group, i.e. of the proteins constituting the complex it belongs to. Also, polymerizations show a similar behaviour: a biological or synthetical polymer-chain can undergo elongation and shortening reactions with a rate independent of the chain length, a scenario suitably described by one rule. This has to be contrasted with the possibly unbounded number of reactions (one for each possible chain length). This representation as rules corresponds to the general notation of organic chemical reactions since parts of molecules that do not take part in a reaction are omitted in reaction equations.

Simulation of rule-based models: Gillespie's Direct Method [4] can be easily applied to simulate efficiently chemical reaction networks represented by reaction rules. Instead of choosing a reaction probabilistically and update the system according to this reaction, one can choose probabilistically a rule, apply this rule to a probabilistically chosen set of reactants and update the system accordingly. While the performance of the Direct Method depends on the num-

SpringSim-MSM, 2015, April 12 - 15, Alexandria, VA, USA

ber of possible different reactions the system can perform, the performance of the simulation of such rule-based models depends on the number of rules that describe the system, which can be exponentially smaller [11]. However, not every chemically reacting system can be easily represented by simple rules as explained above. It is possible that some of the reactions are forbidden under certain conditions. For such systems, an efficient rejection-based algorithm was presented [12, 13]. The choice of the rule and of the reactants works as sketched above. After choosing the reactants, one has to further check whether the chosen reaction is possible or not. If not, the state of the system remains the same, but simulation time is still increased. This approach can also be applied when the model contains reactions with rate constants that are dependent on some global features of the involved reactants [13], like intra-complex enhancement or repression. In this case, the chosen reaction occurs with a probability equal to the ratio of the actual reaction rate, which can be calculated once the reactants have been chosen, and the generic rate of the reaction rule (which is an upper bound of the true rate). Such rejection-based algorithms use the fact that the probabilistic behaviour of CTMCs with added self-loops does not change, as described in [14], and are implemented in state of the art simulators of rule-based models.

Simulation of drug-administration: In many systems, the reaction rates remain constant over time. However, this is often a simplifying assumption. For example, the reaction rate constants could depend on the volume of the considered system (e.g. for bimolecular reactions). Thus, when the volume (for example of cells) changes in time, so does the reaction rate constant. Furthermore, drugs can influence reaction rates with a time-dependent modulation, hence time-varying rates are fundamental for models taking into account *drug administration*. In the Direct Method, time-varying rate constants have a significant influence on the time complexity of the simulation: often the identification of the elapsed time step cannot be computed analytically anymore, but requires the solution of an integral equation. Numerical methods for this task are expensive, like the zero-crossing method used in hybrid simulation [15]. To ameliorate this problem, [16] introduces a rejection-based algorithm to simulate chemical reactions with time-dependent rates. Here the simulation is done within the common Direct Method scheme, replacing the time-dependent rates with an upper bound for their rate constants. When a time-dependent reaction is chosen, the reaction is executed with probability equal to the ratio between the actual reaction rate and its upper bound, at the cost of the generation of an additional uniform random number. In [16], it is proven that this approach samples correctly the traces of a model.

Alzheimer's Disease: The main driver of this paper is the construction of a model of the influence of a γ-secretase inhibitor, a candidate drug on the formation of plaques during Alzheimer's Disease (AD). This system presents two compelling features: (a) the plaque-formation process is a polymerization, hence prone to combinatorial explosion of the underlying reaction network, and (b) the influence of orally administered drugs depends on time. It follows that the two bottlenecks of standard stochastic simulation, large network size and time-dependent rates, manifest themselves at the same time.

Paper contribution and structure: In this paper, we consider how to simulate efficiently this class of stochastic models, providing a simulation routine for rule-based languages with time-varying rates. This constitutes our main methodological contribution. We then put this algorithm at work to simulate the formation of plaques under different drug administration policies, obtaining that splitting the application rate of the drug can lead to a significantly lower concentration of plaques. The paper is organised as follows: Section 2. provides the notation used in the paper, in Section 3. we describe the proposed algorithm and show that it provides correct results and in Section 4. we describe the model of the plaque-formation during AD and show the results of our simulations. Further related work is discussed in Section 5., where conclusions are also drawn.

2. PRELIMINARIES

We will briefly introduce rule-based modelling for biochemical systems, referring the reader to [8, 9] for a more detailed presentation. The basic actors in biochemical networks are molecules, binding sites, and species. A *molecule* often denotes a protein or monomer, and is constituted of *sites*, which can be active or inactive and may bind to each other. In the progress of chemical reactions, many molecules can combine on their binding sites, thus forming complex chemical compounds, constituting the *species* of a system. As binding can happen if a site is active, independently of the actual species the sites belong to, the potential number of species grows exponentially. Even worse, in case of polymerization reactions, the number of species is unbounded (e.g., polymer chains of unbounded length can be formed in principle). Hence, describing biochemical networks by enumerating all reactions involving species is unfeasible.

Rule-based modelling circumvents the problem by focusing on sites, and describing rules involving them, which implicitly generates all possible reactions. More precisely, let $Y = (Y_1, \ldots, Y_K)$ be the vector of all types of binding sites relevant for the reaction network, distinguishing active sites from inactive sites and assuming each site can belong only to a specific molecular type. By $y = (y_1, \ldots, y_K)$ we denote the vector of the counts of active sites from Y in a specific network configuration. When we use a rule-based description, we consider a (finite) set $R = \{r_1, \ldots, r_J\}$ of rules of the form

$$Y_{k_1} + ... + Y_{k_\ell} \xrightarrow{c_{r_j}} Y_{k_{\ell+1}} + ... + Y_{k_L}$$

where r_j refers to the j-th rule ($j \leq J$), c_{r_j} is the associated stochastic reaction rate constant, and $k_1, \ldots, k_L \in \{1, \ldots, K\}$. Species and sites with a stoichiometric coefficient are separated, e. g. $2Y_1$ is written as $Y_1 + Y_1$. Given a list of rules and of sites, we can easily generate all possible species that can be constructed combining molecules according to the rules (if their number is finite), and instantiate the rule by replacing each site with the species it belongs to. Thus, we define $X = (X_1, \ldots, X_N)$ as the vector of all involved species and let $x \in \mathbb{N}_0^N$ be the counts of species $X_1, \ldots, X_N$.[1] Now, we can construct the grounded reaction network [9] as a (possibly infinite) set $S = \{s_1, s_2, \ldots\}$ of reactions of the form

$$X_{n_1} + ... + X_{n_\ell} \xrightarrow{c_{s_i}} X_{n_{\ell+1}} + ... + X_{n_L}$$

where s_i refers to the i-th reaction, c_{s_i} is again the associated stochastic reaction rate constant, and $n_1, \ldots, n_L \in \{1, \ldots, N\}$. The *state* of a system is now completely described by the list (X) and the counts (x) of all species. However, when simulating rule-based models, it is meaningful to update also the counts of all active sites (y) in each step. Usually, the stochastic reaction rate constants c_{r_j} and c_{s_i} are equal. However, c_{s_i} can be further modulated by properties of the species $X_{n_1}, ..., X_{n_\ell}$ that instantiate the rule like chain lengths of polymers or intra-species activations. We will denote such global features generally as ϕ.

When we consider stochastic simulation algorithms, we have to compute the probability that a reaction takes place in a certain time-interval. These probabilities not only depend on the rate constants c_{s_i} or c_{r_j} but also on the numbers of the possible combinations of species $h_{s_i} = f(x)$ or sites $h_{r_j} = f(y)$ that are involved [4]. For instance, for bimolecular reactions $X_1 + X_2$ this number is $x_1 \cdot x_2$. This holds for $X_1 \neq X_2$, for $X_1 = X_2$, $h_{s_i} = 0.5 \cdot x_1 \cdot (x_1 - 1)$. The reaction rate is given by $\alpha_{s_i} = c_{s_i} \cdot h_{s_i}$ and the total rate is $\alpha_S = \sum_{s_i \in S} \alpha_{s_i}$. Similarly, for rule r_j we define $\alpha_{r_j} = c_{r_j} \cdot h_{r_j}$. The total rate is the sum $\alpha_R = \sum_{r_j \in R} \alpha_{r_j}$ and $\alpha_R = \alpha_S$.

3. TIME-DEPENDENT RATES

In the sequel, we propose an algorithm to simulate time-dependent rule-based models of chemical reactions. It combines the rejection-based simulation of rule-based models and the rejection-based simulation of models with time-dependent rate constants. We will start by considering the canonical extension of Gillespie's algorithm for time-dependent rates, with no rejection mechanism, in order to obtain the expression of the correct probability density, that will be used to prove correctness of our rejection-based approach.

[1] We assume N finite for simplicity, w.l.o.g.

3.1. Basic algorithm

We consider a set of rules R with (possibly) time-dependent rate constants. Each of these rules r_j induces a set of different but structurally similar chemical reactions S_{r_j} which means that $\alpha_{r_j}(t) = \sum_{s_i \in S_{r_j}} c_{s_i}(t) \cdot h_{s_i}$. We consider algorithm 1. In an iteration of this algorithm the probability density function that a certain combination of sites undergo a reaction s_i at time τ is readily computed as

$$P(t_0 + \tau) = \alpha_{s_i}(t_0 + \tau) \cdot \exp(-\int_{t_0}^{t_0+\tau} \alpha_R(t) dt). \qquad (1)$$

1: Initialize time t_0, x and y.;
2: **repeat**
3: Store t_0, x and y.;
4: Draw a pseudo random number τ according to the density $F(t_0 + \tau) = \alpha_R(t_0 + \tau) \cdot \exp(-\int_{t_0}^{t_0+\tau} \alpha_R(t) dt)$ and update $t_0 = t_0 + \tau$ [2];
5: Draw a pseudo random number $I \in \{1, 2, \ldots\}$ according to the discrete distribution $\frac{\alpha_{r_1}(t_0+\tau)}{\alpha_R(t_0+\tau)}, \frac{\alpha_{r_2}(t_0+\tau)}{\alpha_R(t_0+\tau)}$, ... , and choose the corresponding rule r_j;
6: Instantiate the chosen rule by probabilistically selecting the species X_{n_1}, ..., the chosen reacting sites Y_{k_1}, ... belong to[3]. The probability to choose the sites according to a reaction s_i is $\frac{\alpha_{s_i}(t_0+\tau)}{\alpha_{r_j}(t_0+\tau)}$;
7: Apply r_j to the species/sites, update x;
8: **until** stop-condition is reached

Algorithm 1: Basic algorithm for simulating time-dependent rule-based models

3.2. Rejection-based algorithm

We now present a rejection-based algorithm that samples the same densitys for all possible reactions as the basic algorithm presented above. Running simulations with the basic algorithm is much more costly[2][3]. As described in the introduction, the main idea is to create null events by simulating the reaction system with higher reaction rates, that are constant in time.

Consider a rule r_j that induces reactions S_{r_j}. For each $s_i \in S_{r_j}$ we define an upper bound $\overline{c_{s_i}}$ of $c_{s_i}(t)$ that is constant in time. Furthermore, we define $\overline{c_{r_j}} = \sup_{s_i \in S_{r_j}} \overline{c_{s_i}}$. Now, we define $\varepsilon_{s_i} = 1 - \frac{c_{s_i}(t)}{\overline{c_{s_i}}}$ and $\overline{\alpha_R} = \sum_{r_j \in R} \overline{\alpha_{r_j}}$ and we can con-

[2] There is no analytical solution for a generic $\alpha_{s_i}(t)$, hence a costly numerical approach is needed.

[3] The dependency of c_{s_i} on global features of the reactants ϕ must be taken explicitly into account, selecting reactant species jointly.

struct algorithm 2. This algorithm combines both rejection-based algorithms we mentioned in the introduction. If a

1: Initialize time t_0, x and y;
2: **repeat**
3: Store t_0, x and y and set $n = 0$;
4: Compute all $\overline{\alpha_{r_j}}$ and $\overline{\alpha_R}$;
5: **repeat**
6: Draw an exponentially distributed pseudo random number τ with rate $\overline{\alpha_R}$ and update $t = t_n + \tau$;
7: Draw a pseudo random number according to the discrete distribution $\frac{\overline{\alpha_{r_1}}}{\overline{\alpha_R}}$, ... , $\frac{\overline{\alpha_{r_J}}}{\overline{\alpha_R}}$ and choose randomly a rule r_j;
8: Instantiate the rule by probabilistically choosing a species X_{n_ℓ} for each reacting site Y_{k_j}, ... [4]and compute for this combination ε_{s_i}. The probability to choose the sites according to a reaction s_i is $\frac{\overline{\alpha_{s_i}}(t_0+\tau)}{\overline{\alpha_{r_j}}(t_0+\tau).}$;
9: Draw a uniformly distributed random number ρ between 0 and 1;
10: set $n = n+1$, $t_n = t.$;
11: **until** $\rho < \varepsilon_{s_i}$;
12: Update x and y according to $r_j.$; Set $t_0 = t_n$;
13: **until** stop-condition is reached;

Algorithm 2: Rejection-based algorithm

rule has a time-dependent rate, all rates of this rule can be set to $\overline{c_{s_i}} = max_t(c_{s_i}(t))$ and ε_{s_i} can be set to $1 - \frac{c_{s_i}(t)}{\overline{c_{s_i}}}$. The same can be done for rate constants that depend on global features of the current state or properties of the probabilistically chosen species ϕ, defining in this case the upper bound as $\overline{c_{s_i}} = max_{t,\phi}(c_{s_i}(t,\phi))$. Note that the global rejection probability can be computed as $\varepsilon_R(t) = \sum_{s_i \in S} \varepsilon_{s_i}(t) \cdot \frac{\overline{\alpha_{s_i}}}{\overline{\alpha_S}}$ and that also $\overline{\alpha_R} = \overline{\alpha_S}$. We now prove the following:

Theorem 1. *The algorithm presented in section 3.2. samples trajectories according to the correct probability density, i.e. the probability density of the next reaction being i and happening at time $t+\tau$ is given by (1).*

We prove this by computing the probability densities of i happening at $t+\tau$ after n rejections and sum up these densities for all n. First an iterated integral is defined as

$$F_1(t_2) = \int_{t_0}^{t_2} \varepsilon_R(t_1) dt_1$$
$$\text{and } F_n(t_{n+1}) = \int_{t_0}^{t_{n+1}} \varepsilon_R(t_n) \cdot F_{n-1}(t_n) dt_n.$$

[4]We can choose X_{r_i} independently for each site, even if the rate constants depend on global features ϕ, as this will be automatically dealt with by the rejection scheme.

Lemma 1. $F_n(t_{n+1}) = \frac{1}{n!}\left(\int_{t_0}^{t_{n+1}} \varepsilon_R(t) dt\right)^n$

Proof. We prove this inductively. By renaming, Lemma 1 holds for $n = 1$ and thus

$$\begin{aligned} F_n(t_{n+1}) &= \int_{t_0}^{t_{n+1}} \varepsilon_R(t_n) \cdot \frac{1}{(n-1)!} \cdot \left(\int_{t_0}^{t_n} \varepsilon_R(t) dt\right)^{n-1} dt_n \\ &\overset{5}{=} \left[\frac{1}{n!} \cdot \left(\int_{t_0}^{t_n} \varepsilon_R(t) dt\right)^n + c\right]_{t_n = t_0}^{t_n = t_{n+1}} \\ &= \frac{1}{n!} \cdot \left(\int_{t_0}^{t_{n+1}} \varepsilon_R(t) dt\right)^n \end{aligned}$$

□

With Lemma 1 we can now prove Theorem 1:

Proof. We start with computing the densities of these reactions firing at time $t_0+\tau$ with no rejection P_0 and exact one rejection at time t_1.

$$P_0(t_0+\tau) = \overline{\alpha_{s_i}} \cdot exp\left(-\overline{\alpha_R} \cdot \tau\right) \cdot (1 - \varepsilon_{s_i}(t_0+\tau)) \quad (2)$$
$$\begin{aligned} P_1(t_0+\tau) &= \int_{t_0}^{t_0+\tau} \overline{\alpha_R} \cdot exp\left(-\overline{\alpha_R} \cdot (t_1 - t_0)\right) \cdot \varepsilon_R(t_1) \\ &\cdot \overline{\alpha_{s_i}} \cdot exp\left(-\overline{\alpha_R} \cdot (t_0 + \tau - t_1)\right) \cdot (1 - \varepsilon_{s_i}(t_0+\tau))\, dt_1 \\ &= F_1(t_0+\tau) \cdot \overline{\alpha_R} \cdot \overline{\alpha_{s_i}} \cdot exp\left(-\overline{\alpha_R} \cdot \tau\right) \cdot (1 - \varepsilon_{s_i}(t_0+\tau)) \end{aligned}$$

Similarly the density for the second rejection at t_2 can be computed as $(\overline{\alpha_R})^2 \cdot F_1(t_2) \cdot exp(-\overline{\alpha_R} \cdot (t_2 - t_0)) \cdot \varepsilon_R(t_2)$ and therefore

$$\begin{aligned} P_2(t_0+\tau) &= F_2(t_0+\tau) \cdot (\overline{\alpha_R})^2 \cdot \overline{\alpha_{s_i}} \\ &\quad \cdot exp\left(-\overline{\alpha_R} \cdot \tau\right) \cdot (1 - \varepsilon_{s_i}(t_0+\tau)) \text{ and} \\ P_n(t_0+\tau) &= F_n(t_0+\tau) \cdot (\overline{\alpha_R})^n \cdot \overline{\alpha_{s_i}} \\ &\quad \cdot exp\left(-\overline{\alpha_R} \cdot \tau\right) \cdot (1 - \varepsilon_{s_i}(t_0+\tau)) \end{aligned} \quad (3)$$

Now the sum of all P_n has to be computed:

$$\begin{aligned} &P(t_0+\tau) = P_0(t_0+\tau) \\ &+ \sum_{n=1}^{\infty} F_n(t_0+\tau) \cdot (\overline{\alpha_R})^n \cdot \overline{\alpha_{s_i}} \cdot exp\left(-\overline{\alpha_R} \cdot \tau\right) \cdot (1 - \varepsilon_{s_i}(t_0+\tau)) \\ &= P_0(t_0+\tau) + P_0(t_0+\tau) \cdot \sum_{n=1}^{\infty} \frac{\left(\overline{\alpha_R} \cdot \int_{t_0}^{t_0+\tau} \varepsilon_R(t) dt\right)^n}{n!} \\ &= P_0(t_0+\tau) \cdot \sum_{n=0}^{\infty} \frac{\left(\overline{\alpha_R} \cdot \int_{t_0}^{t_0+\tau} \varepsilon_R(t) dt\right)^n}{n!} \\ &= \overline{\alpha_{s_i}} \cdot exp\left(-\int_{t_0}^{t_0+\tau} \overline{\alpha_R} dt\right) \cdot (1 - \varepsilon_{s_i}(t_0+\tau)) \\ &\cdot exp\left(\overline{\alpha_R} \cdot \int_{t_0}^{t_0+\tau} \varepsilon_R(t) dt\right) \\ &= \alpha_{s_i}(t_0+\tau) \cdot exp\left(-\int_{t_0}^{t_0+\tau} \alpha_R(t) dt\right) \end{aligned}$$

[5]The primitive can be found by derivation of the needed function.

In the last step we used $\alpha_{s_i}(t) = \overline{\alpha_{s_i}} \cdot (1 - \varepsilon_{s_i}(t))$ and $\alpha_R(t) = \overline{\alpha_R} \cdot (1 - \varepsilon_R(t))$. Both equations follow directly from $c_{s_i}(t) = \overline{c_{s_i}} \cdot (1 - \varepsilon_{s_i}(t))$, for the latter we used that $\alpha_R(t) = \alpha_S(t)$ and $\overline{\alpha_R} = \overline{\alpha_S}$. □

Computational Complexity: The complexity of the proposed rejection sampling algorithm is heavily dependent on the rejection rate for time-dependent transitions, i.e. on the (average) number of rejections per firing of a time-dependent rule. In turn, this depends on how close the upper bound $\overline{c_{r_j}}$ is to the value $c_{r_j}(t)$ of the time-dependent rate at the tentative firing time t, as the cost of performing each step is $O(1)$. A too coarse upper bound can result in many rejections, hence in poor performances. If the time-dependent function(s) heavily fluctuate, we could choose a small time horizon, such that $\overline{c_{r_j}}$ stays in a certain bound. This also means that the approach can be scaled to arbitrary finite time-dependent rates. Practically, in the simulations of the system of Section 4.1, only 1% of the reaction steps were rejected, hence the overload of the rejection scheme is very mild, compared for instance to the simpler simulation algorithm that approximates the time-dependent rate with the value computed at the beginning of each step. Indeed, the number of evaluations of time-dependent functions is considerably smaller in the rejection-based scheme if there are many rules with constant rate (time-dependent rates are evaluated only when a step involving one of them is attempted).

Implementation: An implementation in `C` of the proposed method is available on `http://mosi.cs.uni-saarland.de/?page_id=1341`.

4. RESULTS

We discuss a model of the formation of plaques during Alzheimer's disease under the influence of drugs that shall suppress this formation. First, we explain the model, then we show results we obtained using the presented simulation algorithm.

4.1. A model of fibril production during Alzheimer's disease

Accumulation of amyloid-β (Aβ) peptide in the central nervous system (CNS) is believed to play a crucial role in the pathogenesis of Alzheimer's disease (AD) [17]. Aβ is formed by the enzymes β- and γ-secretase from the amyloid precursor protein (APP) and is produced regularly in the human metabolism. Its original function is unclear but Aβ monomers polymerize until the formation of plaques in the brain. Some hypotheses exist that the intermediate oligomers and not the final plaques are the toxic agents resulting in neurodegeneration[18]. Even if pathogenesis remains unclear, disease modifying strategies for AD include the inhibition of γ-secretase. But the impact of these agents on the production and polymerization of Aβ is unclear. Since it is not possible to measure the Aβ-level in the brain directly, models are needed to get information about the formation of plaques under certain conditions, including disturbing the system by pharmacotherapy. The model we present is based on a deterministic model which takes into account that Aβ not only appears in the brain, but also in the plasma and in the cerebrospinal fluid (CSF) [19]. The model includes production and removal of Aβ as well as transportations of Aβ. In the brain, Aβ can stepwise polymerize and build the plaques. Figure 1 shows a graphical representation of the model. The produced plaques are polymers of proteins and as we stated before, it is unclear if oligomeric intermediates or the final plaques are the toxic agents. Therefore, we want to observe polymer chains of arbitrary lengths and translated the model into a rule-based model.Aβ-proteins in the different parts can then be modelled by different species (M_p, M_b, M_c). The Aβ-species in the brain can react to a dimer (C) that then reacts to a trimer. This trimer is modelled as a chain of length 1 ($C(1)$) with two reacting sites (P), each of them can react with M_b and elongate the chain. The model contains twelve rules that can be displayed as chemical equations as in Figure 2. It is

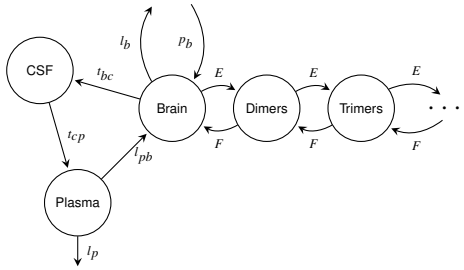

Figure 1. A graphical representation of the model of fibril production.

$$M_p \xrightarrow{l_p} \emptyset \qquad M_b \xrightarrow{t_{bc}} M_c$$
$$M_b \xrightarrow{l_b} \emptyset \qquad M_c \xrightarrow{t_{cp}} M_p$$
$$\emptyset \xrightarrow{p_b} M_b \qquad 2M_b \xrightarrow{2e} D$$
$$M_p \xrightarrow{t_{pb}} M_b \qquad D \xrightarrow{f} 2M_b$$

$$D + M_b \xrightarrow{e} \text{P-C(1)-P}$$
$$\text{C(n)-P} + M_b \xrightarrow{\frac{1}{2}e} \text{C(n+1)-P}$$
$$P - C(1) - P \xrightarrow{f} D + M_b$$
$$C(n+1) - P \xrightarrow{f} C(n) - P + M_b.$$

Figure 2. Reaction scheme

now possible to simulate the system to equilibrium[6]. In [19] the authors then perturbed the system by administering a γ-secretase inhibitor which leads to a reduction of p_b by 40 %. We could perfectly reproduce the results from [19] and we could also obtain data about the distribution of the different chain lengths of the plaques that is not presented in [19].

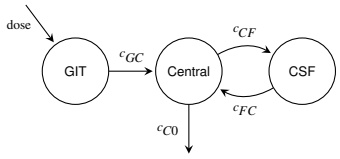

Figure 3. Model of the distribution of the γ-secretase inhibitor

We used the new simulation-algorithm for rule-based models with time-dependent rates to capture the fact that realistic drugs cannot reduce permanently a reaction rate. Moreover, the effect of such drugs changes over time. Since drugs are ingested over the gastro-intestinal tract (GIT) but the γ-secretase inhibitor has an influence on p_b in the brain, we have to model the distribution of the drug over the different compartments. In [20] the authors use the concentration of the γ secretase inhibitor in the central compartment ($[C](t)$) to model p_b as $p_b = p_{b_{max}} \cdot \left(1 - \frac{I_{max} \cdot [C](t)}{IC_{50} + [C](t)}\right)$, where I_{max} is the maximal inhibitory effect of the drug and IC_{50} is the half maximal inhibitory concentration. To compute $[C](t)$, we use the model from [20] which is shown in Figure 3. The γ-secretase inhibitor is given to the GIT (dose) and it is absorbed to the central compartment (c_{GC}). It is then distributed (with c_{CF}/c_{FC}) between the central compartment and the cerebrospinal fluid (CSF) and cleared from the central compartment (c_{C0}).

For our model, we used the equations from Figure 4, we fixed I_{max} and IC_{50} and adjusted the dose of the γ-secretase inhibitor in a way that p_b is reduced by 40% in average. We ran simulations of the system by assuming dosing intervals of 8 and 24 respectively, hours[7]. Simulating the model for 24 hours (model time) took a computational time of 91 seconds on one 2.0 GHz processor. In Figure 5 and 6 some of the results of these simulations are shown. It is obvious that the dosing interval has an influence on the concentrations of the plaques in the brain. Splitting the daily dose into three doses has a big influence on the concentration of the short polymer chains (factor of 2.5 between both settings after 460 days, Figure 5). We also show the influence of the splitting of the daily dose on the overall distribution of the chain length after

[6]We used the rate constants from [19], $(l_p, l_b, p_{b_{max}}, t_{pb}, t_{bc}, t_{cp}, e, f) = (6.7 \cdot 10^{-3}, 0.278 \cdot 10^{-3}, 7.34 \cdot 10^{-12}, 37.7 \cdot 10^{-6}, 7.6 \cdot 10^{-6}, 1.7 \cdot 10^{-3}, 90, 1.38 \cdot 10^{-6})$

[7]For a dosing interval of 24 h, we assumed the dose to be 91.92 mg, for the dosing interval of 8 h, we assumed 30.64 mg.

$$\frac{d[GIT]}{dt} = \begin{cases} [dose] - c_{GC} \cdot [GIT] & \text{, if } t = n \cdot [dosing\_int] \\ -c_{GC} \cdot [GIT], \text{ else} \end{cases}$$

$$\frac{d[CSF]}{dt} = c_{CF} \cdot [C] - c_{FC} \cdot [CSF]$$

$$\frac{d[C]}{dt} = c_{GC} \cdot [GIT] + c_{FC} \cdot [CSF] - c_{CF} \cdot [C] - c_{C0} \cdot [C]$$

$$p_b = p_{b_{max}} \cdot \left(1 - \frac{I_{max} \cdot [C](t)}{IC_{50} + [C](t)}\right)$$

$c_{GC} = 0.624h^{-1}$

$c_{CF} = 0.000227h^{-1}$ $\quad I_{max} = 1$

$c_{FC} = 0.573h^{-1}$ $\quad IC_{50} = 32ng \cdot mL^{-1}$

$c_{C0} = 0.253h^{-1}$

Figure 4. computation of p_b

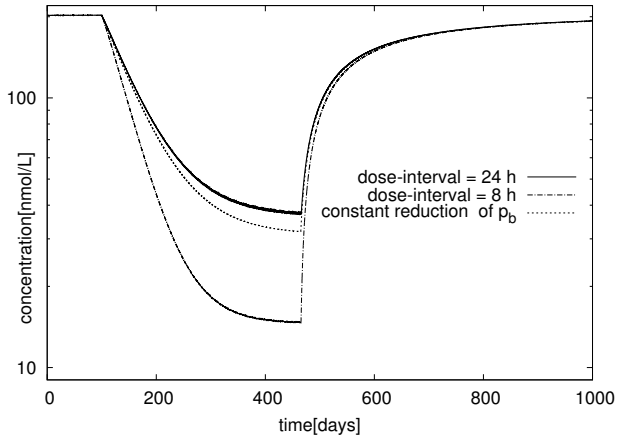

Figure 5. Concentration of the plaques with length 2 to 9. The simulation started in the equilibrium, after a simulation time of 100 days, the taking of the γ-secretase inhibitor was simulated by reducing p_b according to the differential equations from Figure 4. After 465 days the taking of the medicament was stopped. Additionally results from simulations according to [19] are shown.

460 days. The concentrations of all chain lengths is significantly lower when the daily dose was split (Figure 6).

4.2. Comparison to a simple approach

In Section 3.2., we mentioned a simpler approach to simulate reactions with time-dependent rates by approximating the rates in each simulation step. We claim that this approximation can lead to significantly different results for some models. In order to validate this claim, we consider a scenario in which the rate function increases quickly in time while few

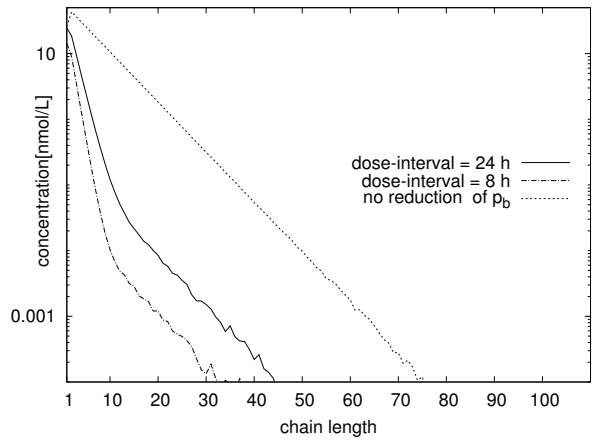

Figure 6. Distribution of the chain length of the Aβ chains after 460 days. Here the results are compared to the distribution of the chain length in the equilibrium.

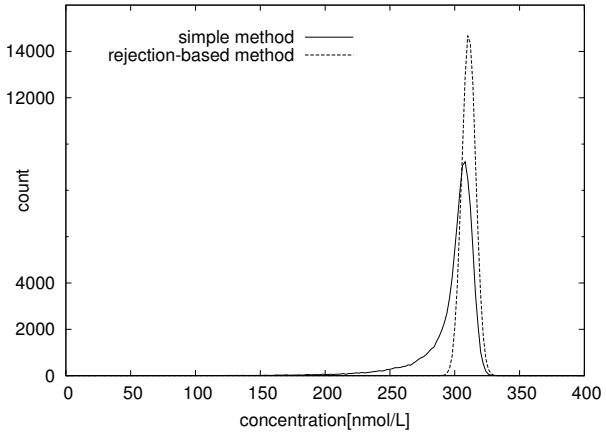

Figure 7. Distribution of the concentration of the monomers in the brain after 10 seconds. We used two approaches and a modified model.

reactions fire overall. Specifically, we set in the simulation of the model from Section 4.1. $p'_{b_{max}} = 1000 \cdot p_{b_{max}}$ and in the computation of the time-dependent function we set all constants c_{XY} to $c'_{XY} = 48 \cdot c_{XY}$ and $p'_b = p'_{b_{max}} \cdot \left(\frac{I_{max} \cdot [C](t)}{IC_{50} + [C](t)}\right)$ (i.e. at $t = 0$ the system is in the equilibrium and p'_b is suddenly set to $p_b = 0$). Figure 7 shows the distribution of the concentrations of the monomers in the brain after simulating the system 100000 times with both approaches for 100 seconds. It can be seen that the resulting distributions are quite different (the two sided Kolmogorov-Smirnov test yields a p-value of $p < 10^{-32}$). Since we proved that the rejection-based method is correct, we conclude that the simple approach fails here. This happens because at the beginning of the simulation the overall rate is very low, while the rate of the time-dependent event is zero. Hence, in the simple model, this transition can fire only after another event happens. However, the time-dependent rate increases fast, hence we expect this event to fire much sooner than as predicted by the simple method. Therefore, the production of monomers is delayed in the simple method and the associated peak in Figure 7 has a shoulder on the left side. However, as our rejection-based method incurs in a very moderate overhead, it should always be preferred.

5. RELATED WORK AND CONCLUSION

Simulation methods based on rejections are known for a long time. In the context of non-homogeneous poisson processes, a methodical overview was already given in the late seventies [21]. Also, in newer days, many approaches with rejection-based simulation algorithms were presented. In addition to the previously mentioned rejection-based simulation approaches of rule-based models [12, 13] and the rejection-based simulation of models with time-dependent rates [16] a general rejection-based simulation algorithm for stochastic models of chemical reaction systems is presented in [22]. Here the authors gain simulation speed by avoiding the recomputation of the actual propensities by exploiting rejections. Other simulations of models with time-dependent rate constants are mostly done by variants of the Next Reaction Method [23, 6]. The research in the field of Alzheimer's Disease is widely spread. One branch is engaged in the amyloid hypothesis and possibilities to find therapeutics against AD within this setting. Some general work on this can be found in [17] and [24]. More specialised work can, for example, be found in [25, 26] with respect to γ-secretase inhibitors and in [18] about the toxicity of different Aβ-oligomers.

We presented a general method to simulate rule-based models with time-dependent rate constants. We proved the correctness of the approach and presented results of the simulation of a model of the formation of plaques during AD. We showed that with our approach it is possible to model the influence of different doses and dosing intervals of drugs that prevent the formation of plaques. In our future work, we want to use these insights to provide an updated version of the presented model and to analyse further strategies for AD Research and Drug Development.

ACKNOWLEDGEMENT: This work was partially funded by the German Research Council (DFG) as part of the Cluster of Excellence on Multimodal Computing and Interaction at Saarland University and the Transregional Collaborative Research Center SFB/TR 14 AVACS.

REFERENCES

1. H. H. McAdams and A. Arkin. Stochastic mechanisms in gene expression. *PNAS*, 94(3):814–819, 1997.

2. P. S. Swain, M. B. Elowitz, and E. D. Siggia. Intrinsic

and extrinsic contributions to stochasticity in gene expression. *PNAS*, 99(20):12795–12800, 2002.

3. H. Tobita. Simulation model for the modification of polymers via crosslinking and degradation. *Polymer*, 36(13):2585–2596, 1995.

4. D. T. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of Computational Physics*, 22:403–434, 1976.

5. D. T Gillespie. Exact stochastic simulation of coupled chemical reactions. *The journal of physical chemistry*, 81(25):2340–2361, 1977.

6. M. A. Gibson and J. Bruck. Efficient exact stochastic simulation of chemical systems with many species and many channels. *The journal of physical chemistry A*, 104(9):1876–1889, 2000.

7. M. Rathinam, L. R. Petzold, Y. Cao, and D. T Gillespie. Stiffness in stochastic chemically reacting systems: The implicit tau-leaping method. *The Journal of Chemical Physics*, 119(24):12784–12794, 2003.

8. V. Danos, J. Feret, W. Fontana, R. Harmer, and J. Krivine. Rule-based modelling of cellular signalling. *Lect Note Comput Sci*, 4703:17–41, 2007.

9. J. R. Faeder, M. L. Blinov, B. Goldstein, and W. S. Hlavacek. Rule-based modeling of biochemical networks. *Complexity*, 10:22–41, 2005.

10. F. RD van Parijs, K. Morreel, J. Ralph, W. Boerjan, and R. MH Merks. Modeling lignin polymerization. i. simulation model of dehydrogenation polymers. *Plant physiology*, 153(3):1332–1344, 2010.

11. M. W. Sneddon, J. R. Faeder, and T. Emonet. Efficient modeling, simulation and coarse-graining of biological complexity with nfsim. *Nature Methods*, 8:177–185, 2011.

12. V. Danos, J. Feret, W. Fontana, and J. Krivine. Scalable simulation of cellular signaling networks. In *Programming Languages and Systems*, pages 139–157. Springer, 2007.

13. J. Yang, M. I. Monine, J. R. Faeder, and W. S. Hlavacek. Kinetic monte carlo method for rule-based modeling of biochemical networks. *Physical Review E*, 78(3):031910, 2008.

14. C. Baier, J.-P. Katoen, H. Hermanns, and V. Wolf. Comparative branching-time semantics for markov chains. *Information and computation*, 200(2):149–214, 2005.

15. F. Zhang, M. Yeddanapudi, and P. Mosterman. Zero-crossing location and detection algorithms for hybrid system simulation. In *IFAC World Congress*, pages 7967–7972, 2008.

16. V. Holubec, P. Chvosta, M. Einax, and P. Maass. Attempt time monte carlo: An alternative for simulation of stochastic jump processes with time-dependent transition rates. *EPL (Europhysics Letters)*, 93(4):40003, 2011.

17. D. J. Selkoe. Toward a comprehensive theory for alzheimer's disease. *Annals of the New York Academy of Sciences*, 924(1):17–25, 2000.

18. I. Benilova, E. Karran, and B. De Strooper. The toxic Aβ oligomer and alzheimer's disease: an emperor in need of clothes. *Nature neuroscience*, 15(3):349–357, 2012.

19. D. L. Craft, L. M. Wein, and D. J. Selkoe. A mathematical model of the impact of novel treatments on the aβ burden in the alzheimers brain, csf and plasma. *Bulletin of mathematical biology*, 64(5):1011–1031, 2002.

20. K. G. Haug, A. Staab, C. Dansirikul, and T. Lehr. A semi-physiological model of amyloid-β biosynthesis and clearance in human cerebrospinal fluid. *The Journal of Clinical Pharmacology*, 53(7):691–698, 2013.

21. P. A. Lewis and G. S. Shedler. Simulation of nonhomogeneous poisson processes by thinning. *Naval Research Logistics Quarterly*, 26(3):403–413, 1979.

22. V. H. Thanh, C. Priami, and R. Zunino. Efficient rejection-based simulation of biochemical reactions with stochastic noise and delays. *The Journal of chemical physics*, 141(13):134116, 2014.

23. D. F. Anderson. A modified next reaction method for simulating chemical systems with time dependent propensities and delays. *The Journal of chemical physics*, 127(21):214107, 2007.

24. J. Hardy and D. J. Selkoe. The amyloid hypothesis of alzheimer's disease: progress and problems on the road to therapeutics. *Science*, 297(5580):353–356, 2002.

25. R. J. Bateman, L. Y. Munsell, J. C. Morris, R. Swarm, K. E. Yarasheski, and D. M. Holtzman. Human amyloid-β synthesis and clearance rates as measured in cerebrospinal fluid in vivo. *Nature medicine*, 12(7):856–861, 2006.

26. R. J. Bateman, E. R. Siemers, K. G. Mawuenyega, G. Wen, K. R. Browning, W. C. Sigurdson, K. E. Yarasheski, S. W. Friedrich, R. B. DeMattos, P. C. May, et al. A γ-secretase inhibitor decreases amyloid-β production in the central nervous system. *Annals of neurology*, 66(1):48–54, 2009.