Studying Steady States in Biochemical Reaction Systems by Time Petri Nets

Louchka Popova-Zeugmann[1] and Elisabeth Pelz[2]

[1] Department of Computer Science, Humboldt University, Berlin, Germany,
`popova@informatik.hu-berlin.de`
[2] LACL, University Paris Est Créteil, Fac de Sciences, F- 94010 Créteil
`pelz@u-pec.fr`

Abstract. Biochemical reaction systems are usually modeled by ordinary differential equations (ODEs). For further analysis, they are often transformed into stochastic Petri nets (SPN), whose state space (or reachability graph) then can be studied to deduce properties.

If a biochemical reaction system is in a steady state, from now on called **steady situation**,[1] then the rates of the reactions and the concentrations of the species are constant. These concentrations and rates can be established by simulation of the SPN-model. A steady situation,[1] signifies also that on the model level only a subset of all possible reachable states is pertinent to this situation. It would be of interest to isolate formally and constructively this subset of states. To our knowledge there is no way to achieve this using the SPN-model or the ODE-model.

In this article we propose an approach to calculate the part of the state space corresponding to a steady situation[1]. To do so, we map the SPN-model onto a Time Petri Net-model (TPN) with the same behaviour as that in the steady situation[1] observed in the SPN simulation.

Using reduction methods for TPNs we can extract the part of the reachability graph of the SPN-model which is relevant for the steady-situation[1]. We show that this is exactly the reduced reachability graph of the constructed TPN-model. Finally, the later one can be analyzed qualitatively and quantitatively.

In addition, this approach helps for validating the correctness of the calculated (and used) rates in the steady situation[1]and of the parameters used in the original ODEs, fixed by experiments in the wet labs, both being -a priori- subject to a certain degree of uncertainty.

1 Introduction

When considering biochemical reaction systems we emphasize the interactions between different species during time and we do not take a momentary snapshot of the system. This means that time is an indispensable component in each

---

[1] to avoid confusion, between **state** in the biochemical system and **states** or **state space** in the models, the biological **steady state** will be called throughout the whole paper **steady situation**

model of such a system. Furthermore, the repetitive occurrence of reactions in the system during a certain time, expressed by their reaction rates, defines the behaviour of the system. It is obvious, that the rates depend on the concentrations of the species involved in the reactions: the higher the concentration the higher the reaction rate. This is the case until the concentrations of the involved species achieve certain levels. Then the concentrations of the species do no longer change, i.e., the reaction rates stay constant. This situation is the so called steady situation[1] in an biochemical reaction system. Finally, the occurrence (or taking place) of biochemical reactions is a stochastical one.

The taking place of a particular reaction can be modeled by an ordinary differential equation (ODE), the causal relationship between the interactions is often modeled by some graph. Both aspects can be represented in a unique model, by using some variant of Petri Nets, such as e.g. Hybrid Functional Petri Nets [9,10] or Continuous Petri Nets [16] or Stochastic Petri Nets (SPNs) [8]. The last ones model the stochastic nature of a reaction system especially well. The ODE model can be obtained by means of punctual measured data and using interpolation, cf. [4], or from a first established Modular Interaction Network, cf. [18]. The SPN model, as graph models in general, can be obtained from a system of ODEs which describes the reaction system; such translations are well explained in e.g. [6,9,10]. In general, a system of ODEs defines a unique SPN but it is possible that different systems of ODEs define the same SPN. Conditions for one-to-one and onto mappings between ODEs and SPNs are given in [16].

Please note that an essential point while constructing an SPN-model is the definition of its initial marking. It should faithfully map the initial concentrations of all species involved in the reaction system.

Considering the models quoted above, it is not possible by formally analyzing it, to extract the steady situation[1] in which the reaction system may stay after some time. The only thing which can be done, and which is done in general, is to simulate the model (over a very high number of runs) until being able to deduce properties concerning the steady situation[1], with some remaining uncertainty.

In this paper we are using the uncertain data obtained by simulation of the SPN, and also uncertain parameters estimated by measures and interpolation, and prove analytically if they present in fact those of a steady situation[1]. For this reason, we first observe the SPN-model during a high number of runs which yield mean values corresponding to the concentration of species and reaction rates in the steady situation[1]. In function of these values, we map the SPN onto a Time Petri Net-model (TPN), having the same skeleton, such that the behaviour (that of the observed steady situation[1]) stays the same.

This works, because TPNs have the same semantics as SPNs with constant rates, and we dispose of a well established theory [11–13] for studying analytically their behaviour. In particular, our reduction results concerning state spaces of TPNs [12,13] will be of good use in the presented work. When the simulated reaction rates in the steady situation[1] are exactly the rates in the real situation, then the reduced reachability graph of the TPN should consist of cycles only - up to some initiation part. By contraposition we may conclude, that a non

cyclic form of the reduced state space indicates severe problems in the set of data used to build the TPN, and by consequence, in the initially established data from experiences in the wet labs. Furthermore, we are able to calculate the time-length of the cycle(s), a data which cannot be measured within the SPN model. The last one can be compared with measures from the wet labs, if there are any. Thus once more, we bridge back to the original data. Thus our method offers a way of validating a complex modeling process of biochemical reaction systems.

This paper is organized as follows: in the next section we recall some basic notions and notations of the used Petri Net classes together with the reduction results of TPN state spaces. In section 3 we introduce the mapping from an SPN in the steady situation[1] onto a TPN model. In the subsequent section we illustrate our approach on the core model of the influence of the Raf-1 Kinase Inhibitor Protein (RKIP) on the Extracellular signal Regulated Kinase (ERK) signalling pathway, chosen as running example, before concluding.

2 Basic Concepts

In this section we recall the concept of TPNs. After that we introduce some basic notions and fundamental properties, which are important for their quantitative and qualitative evaluation.

2.1 Basics

Time Petri Nets (TPN) [11] are derived from classical Petri nets by assigning to each transition t a (continuous) time interval $[a_t, b_t]$. Here a_t and b_t are relative to the time when t was enabled most recently. When t becomes enabled, it can not fire before a_t time units have elapsed, and it has to fire not later than b_t time units, unless t got disabled in between by the firing of another transition. The firing itself of a transition does not consume time. So, the given time intervals specify reaction times for the transition firings. The time intervals are defined on non-negative real numbers, but the interval bounds are given as nonnegative rational numbers. Rational numbers are sufficient to reflect any measuring accuracy required by a given application domain. Moreover, to support the normalization of different time scales within a model, zero and ∞ are allowed as interval bounds.

As usual, in this paper, $\mathbb{N}$ denotes the set of natural numbers, and $\mathbb{Q}_0^+$, resp. $\mathbb{R}_0^+$, the sets of nonnegative rational numbers, resp. real numbers. T^* denotes the set of all finite words over the alphabet T, $l(w)$ is the length of a given word w.

Some 5-tuple $\mathcal{Z} = (P, T, v, m_o, I)$ is called a **Time Petri net (TPN)**, if $S(\mathcal{Z}) := (P, T, v, m_o)$, the **skeleton** of $\mathcal{Z}$, is a Petri net where P, T are finite sets with $P \cap T = \emptyset$, $v : (P \times T) \cup (T \times P) \longrightarrow \mathbb{N}$ defines the arcs with their weight, $m_o : P \longrightarrow \mathbb{N}$ fixes the initial marking, and $I : T \longrightarrow \mathbb{Q}_0^+ \times (\mathbb{Q}_0^+ \cup \{\infty\})$ is its

interval function where $\forall t \in T$, $I(t) = [I_1(t), I_2(t)]$ and $I_1(t) \leq I_2(t)$, spezifying the **earliest** and **latest firing time of** t: $eft(t) = I_1(t)$, $lft(t) = I_2(t)$.

As shown in [11], considering TPNs with $I : T \longrightarrow \mathbb{N} \times (\mathbb{N} \cup \{\infty\})$ will not result in a loss of generality. Therefore, only such time functions I will be considered subsequently.

A marking $m : P \longrightarrow \mathbb{N}$ can be seen as a vector of size $|P|$, we refer to it as *p*-**marking**. Thus each transition $t \in T$ induces the *p*-markings t^-, t^+ and Δt defined by $t^-(p) := v(p, t)$, $t^+(p) := v(t, p)$ and $\Delta t(p) := t^+(p) - t^-(p)$. With these notions the **firing rule** for TPNs can be defined. A transition $t \in T$ is **enabled** at a marking m iff $t^- \leq m$ (e.g. $t^-(p) \leq m(p)$ for every place $p \in P$).

The **pre-sets** and **post-sets** of a place or transition x are given by $^\bullet x := \{y \mid v(y, x) > 0\}$ and $x^\bullet := \{y \mid v(x, y) > 0\}$, respectively.

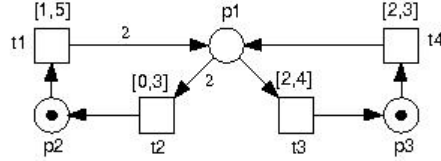An example for an arbitrary TPN is shown in Fig. 1.

Fig. 1. A Time Petri net $\mathcal{Z}_1$.

Every possible situation in a given TPN can be described completely by a **state** $z = (m, h)$, consisting of a *p*-marking m (the standard marking) and a transition-marking (short: t-marking) h. The **t-marking** is a transition vector, which describes the current time circumstances in a certain situation. More exactly, each component of the t-marking is either a real number or the sign $\sharp$. Thus $h(t)$ can be seen as clock of t. If t is enabled at a marking m, its clock $h(t)$ shows the time elapsed since t became most recently enabled. If t is disabled at m, the clock is switched off (indicated by $h(t) = \#$).

Formally, a pair $z = (m, h)$ with $m : P \longrightarrow \mathbb{N}$ and $h : T \longrightarrow \mathbb{R}_0^+ \cup \{\#\}$ is called a **state** of a TPN $\mathcal{Z} = (P, T, v, m_o, I)$ if $\forall t \in T$, either $(t^- \leq m$ and $h(t) \leq lft(t))$ or $(t^- \not\leq m$ and $h(t) = \#$).

The **initial state** $z_o := (m_o, h_o)$ of the TPN $\mathcal{Z}$ is given by defining h_o as follows

$$\forall t \in T, h_o(t) := \begin{cases} 0 & \text{if } t^- \leq m_0 \\ \# & \text{if } t^- \not\leq m_0. \end{cases}$$

Thus, the initial state of $\mathcal{Z}_1$, as given in Fig. 1, is $z_0 = (\underbrace{(0, 1, 1)}_{p\text{-marking}} , \underbrace{(0, \sharp, \sharp, 0)}_{t\text{-marking}})$.

The state $z = (m, h)$ is called an **integer state**, if $h(t)$ is an integer for each enabled transition t in m.

The behaviour of a TPN is defined by changing from one state into another by **firing a transition** (without auto-concurrency) or by **time elapsing**. In

order to define these dynamic aspects of TPNs we need first the notion **ready to fire**.

Let t_o be a transition in T and $z = (m, h)$ a state of a TPN $\mathcal{Z} = (P, T, v, m_o, I)$. The **transition t_o is ready to fire** at state z, denoted by $z \xrightarrow{t_o}$, if $t_o^- \leq m$ and $eft(t_o) \leq h(t_o)$.

Then the **state z change** into a state $z' = (m', h')$ by the **firing** of such a t_o, denoted by $z \xrightarrow{t_o} z'$, where the new marking is $m' = m + \Delta t_o$ and the new clock satisfies $\forall t \in T, h'(t) := \begin{cases} \# & \text{if } t^- \not\leq m' \\ h(t) & \text{if } t \neq t_0, \ (t^- + t_o^-) \leq m, \ t^- \leq m' \\ 0 & \text{otherwise.} \end{cases}$

This definition implies, that in the case that t_0 is still enabled after the firing of t_0, it can only refire after at least α_t new waiting units. To resume, concurrency but no auto-concurrency is possible by the way the evolving of the clocks is defined.

The **state z** may also **change** into a state $z' = (m, h')$ **by the time elapsing** $\tau \in \mathbb{R}_0^+$, denoted by $z \xrightarrow{\tau} z'$, where the marking stays the same, but time goes on : $\forall t \in T$ with $h(t) \neq \#$ we need $h(t) + \tau \leq lft(t)$) i.e. the time elapsing τ need to be possible, and the new clock is given $\forall t \in T$ by
$$h'(t) := \begin{cases} h(t) + \tau & if \ t^- \leq m' \\ \# & if \ t^- \not\leq m'. \end{cases}$$

A state $z = (m, h)$ of a TPN $\mathcal{Z}$ is called **reachable** in $\mathcal{Z}$ (starting at z_0), if there exist states $z_1, z_1', ..., z_n, z_n'$, transitions $t_1, ..., t_n$, and times $\tau_i \in \mathbb{R}_0^+$, for $i \leq n$, such that $z_0 \xrightarrow{\tau_0} z_1 \xrightarrow{t_1} z_1' \xrightarrow{\tau_1} z_2 \xrightarrow{t_2} z_2' \xrightarrow{\tau_2} \dots z_n \xrightarrow{t_n} z_n' \xrightarrow{\tau_n} z$ holds.

The **sequence of transitions** $\sigma = t_1 \dots t_n$ leading to a reachable state will be called a **feasible** one (starting at z_0) or just a **firing sequence** of $\mathcal{Z}$. The full sequence $\sigma(\tau) = \tau_0 t_1 \tau_1 \dots t_n \tau_n$ is called a **(feasible) run** of σ. It shows that in a given TPN the state changes generally consist of alternating series of time elapsing and transition firing. Obviously, for a given run the transition sequence is well defined, and for a given firing sequence there are infinitely many runs in general.

Eventually, $RS_{\mathcal{Z}}(z')$ is the **set of all reachable states** in $\mathcal{Z}$ starting from an arbitrary state z'. And $RS_{\mathcal{Z}} := RS_{\mathcal{Z}}(z_0)$, that from the initial state is also called the **state space** of $\mathcal{Z}$.

We may also consider the **set of reachable p-markings**, also called the **$p-$marking space** $R_{\mathcal{Z}} := \{ m \mid (m, h) \in RS_{\mathcal{Z}}\}$ in a TPN $\mathcal{Z}$. This is a subset (not necessarily proper) of the reachable markings of the skeleton $S(\mathcal{Z})$. Therefore, a firing sequence in the skeleton $S(\mathcal{Z})$ is *not necessarily* a firing sequence in $\mathcal{Z}$. The set of p-markings, reachable in $\mathcal{Z}$ starting at an arbitrary p-marking m', is denoted by $R_{\mathcal{Z}}(m')$. A TPN is called **bounded**, if its set of reachable p-markings is finite, otherwise it is called **unbounded**.

For different reasons the state space of a TPN is in general infinite and dense in terms of the time: the set of reachable p-markings can be infinite or the set of t-markings for a fixed p-marking can be infinite or both together. Later on, we

consider some approaches for concise state space representations, when $RS_{\mathcal{Z}}$ is infinite while $R_{\mathcal{Z}}$ is finite.

The definition of state change by time elapsing can be slightly and consistently modified for the introduction of a reachability graph based on all **reachable essential states** for arbitrary TPNs, especially for TPNs including transitions whose lfts are ∞. The set of all **reachable essential states** for arbitrary TPNs is defined as a subset of all reachable integer states of the considered TPN. We will use the following property (for more details see [13]): if no transition t in $\mathcal{Z}$ has $lft(t) = \infty$ then the set of essential states is exactly the set of integer states in $RS_{\mathcal{Z}}$.

2.2 Time-dependent Minimal and Maximal Runs in TPNs

We will formalize the time-dependent notions of measuring the length of runs, cf. [15].

Let $\sigma(\tau)$ be a run of the transition sequence σ in some TPN $\mathcal{Z}$. The **length of the run** $l\big(\sigma(\tau)\big)$ is the sum of all times while executing the run $\sigma(\tau)$, i.e.,

$$l\big(\sigma(\tau)\big) := \sum_{i=0}^{n} \tau_i, \quad \text{where} \quad n = l(\sigma) \quad \text{and} \quad \tau = \tau_0 \tau_1 \ldots \tau_n.$$

For a given transition sequence σ in $\mathcal{Z}$, a feasible run $\sigma(\tau)$ with minimal length will be referred to as **minimal run** of σ. Evidently, it satisfies :

$$l\big(\sigma(\tau)\big) := \min_{\tau'}\{\, l\big(\sigma(\tau')\big) \mid \sigma(\tau') \text{ is a feasible run of } \sigma \text{ in } \mathcal{Z} \,\}.$$

The notion of **maximal run** can be introduced analogously. It denotes the run with maximal length within all feasible runs of σ if such an upper bound exists; otherwise it is not defined.

The notions of **minimal**, respectively **maximal time distance between two states** can be found in [12, 15] and are useful for precise analysis of biochemical reaction systems which present different kind of steady situations[1] than our current running example.

2.3 State Space Reduction

The central problem for the dynamic analysis of a given TPN is the adequate knowledge of its state space. It is important to get a finite description of the infinite state spaces, under the condition that the p-marking space is finite.

It can be shown that - despite the continuous nature of the time intervals - it is sufficient to pick up just some "essential" states to determine the entire

timed behaviour of the net so that qualitative and quantitative analyses remain possible.
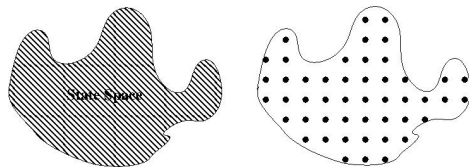

Fig. 2. For some TPN $\mathcal{Z}$. Left hand. Sketched state space of $\mathcal{Z}$: a continuous set. Right hand. Sketched reduced state space of $\mathcal{Z}$: all reachable integer states.

While the calculation of a single reachable integer state is rather straightforward, the proof that *the knowledge of the integer states is sufficient for analyzing a TPN* was quite difficult. Three solutions had been proposed in the past: considering a global clock [11] or considering a parametrical description of the state space [15] or dividing into a finite number of problems, which can be solved recursively with a methodology inspired from dynamic programming [12]. As result, one is able to construct for each TPN a **reduced reachability graph** whose vertices are the **essential states**. When the TPN does not contain a transition whose lft is ∞, then the essential states are exactly all the reachable integer states in the net.

An edge (z_1, z_2) labeled (k, t), with $k \in \mathbb{N}$, in this reduced reachability graph has the meaning that in state z_1 k time units are elapsing before transition t fires leading to state z_2. Now, for finding minimal and maximal time paths between two states/p-markings in a TPN, its reduced reachability graph can be used, even effectively. Our algorithms for computing the reduced reachability graph of a given TPN are implemented in several standard Petri Net tools, like INA [17], tina [3] and charlie [7]. INA can additionally compute minimal and maximal time-dependent paths. Thus these tools can be successfully applied for models of bio-chemical reaction systems, too.

2.4 Stochastic Petri nets

Stochastic Petri Nets (SPNs) had been introduced at the beginning of the Eighties, cf. [1, 2]. They are widely used in the modeling of biochemical reaction systems, cf. [8].

Such SPNs are derived from classical PNs by assigning to each transition t a firing rate λ_t. This firing rate specifies a firing delay for the transition. More exactly, the firing delay is a random variable which is distributed exponentially

and has λ_t as parameter of the probability density function. In fact, to each transition t a probability density function with parameter λ_t is associated :

$$f_t(x; \lambda_t) = \begin{cases} \lambda_t e^{-\lambda_t x}, & x \geq 0, \\ 0, & x < 0. \end{cases}$$

Finally, the firing rate λ_t may be marking-dependent in general. In such a case, we should write $\lambda_t(m)$, where m is a marking, instead of λ_t. Than the expected value for the firing delay for the transition t in the marking m is $\frac{1}{\lambda_t(m)}$.

The firing mode is defined as follows: In a given marking m, each enabled transition t obtains an instance of the firing delay $\lambda_t(m)$ from its associated probability density function. Then a choice is made: the transition with the minimum firing delay is firing. The firing itself of a transition does not consume time. The successor marking is than obtained as in the underlying classical PN. It is well known [1], that the probability for two transitions to fire at the same instant is null, i.e. there is no conflict. That is why the transitions in SPNs fire naturally one by one, i.e., just as in TPNs.

3 Biochemical Systems and Time Petri Nets

Biochemical reaction networks are mostly described by ordinary differential equations (ODEs) or reaction rate equations (RREs), and both can be converted into each other. Taking account of the rate equations of all reactions in the systems, ODE like RRE models can be transformed into Continuous Petri nets or Stochastic Petri nets. More about these transformations can be found, e.g. in [8]. Conditions for a uniform transformation of ODEs into Continuous Petri nets (or RREs) are introduced in [16]. Systems of ODEs can be represented as hybrid functional Petri nets, cf. [9, 10], too. These Petri net models allow for qualitative and quantitative evaluations using tools and methods of the Petri net theory, cf. [2, 5].

A transformation of an ODE model into a Time Petri net model (TPN) using the reaction rates is shown in [14]. This transformation allows the computation of time-minimal and time-maximal paths (if existing) between two system situations, i.e. two states of the TPN model. It can be considered as an indication for the conformance and coherence of the model if the length of the time-minimal and time-maximal paths coincide with the results in the wet labs. Otherwise the original model becomes invalidated.

Independently from the original model, an RRE one or an ODE one, in a first step, a timeless Petri net is always derived. This describes the causal relations between the events in the system. In biochemical systems, these are biochemical reactions or biochemical signal transductions. Thereafter additional information, in particular the time parameters, need to be assigned to the Petri net. They are obtained from the parameters (kinetic rate constants) in the ODEs. Their values are often determined experimentally. When it is not possible to collect

or identify them in vitro, the parameters are estimated using experimental data achieved only for some discrete time points. In this case the goal is to estimate the value of the parameters for each moment so that the values over the time fit the experimental data (cf. [9]). Thus in a second step, integrating these parameters, a time-dependent PN model is established for the biochemical network. It is obviously that at this stage of modeling a certain level of inexactness is present in each model.

In this paper we are going to study biochemical reaction systems which possess a steady situation[1]. This is the case when the system comes in a situation, in which the concentration of all substances stays constant. Usually, in the steady situation[1] the concentration of all substances allows that all reactions take place permanently. Now constructing the reachability graph, may be interpreted as considering the path of changes of the single substances. Loosely speaking, we should get a cyclic set of states in the reachability graph corresponding to the steady situation[1]. The behaviour of the SPN, expressed by Markow chains is isomorphic to the full reachability graph of the underlying PN, i.e., they have the same state space. By convention, we speak in the following of "the reachability graph" of the SPN. But the nodes corresponding to the steady situation[1] **can not** be recognized in this reachability graph, even knowing the rates in the steady situation[1]. To our knowledge, no method is known until now for separating or extracting the subgraph corresponding to the steady situation[1] from the reachability graph of the SPN. Steady state meaning cyclic behaviour, this subgraph (of the reachability graph of the SPN) is supposed to present a cyclic structure (with one or more circles), up to some initiation part. In contrast, knowing the steady situation[1] rates we are able to separate the reachable states, we are looking for, using a TPN and its reachability graph. This is due to the reduction results on reachability graphs of TPNs discussed in section 2.3.

Thus, we propose in this paper a methodologie to calculate and verify such set of states which correspond to steady situation[1]. The starting point will always be an SPN model for a biochemical reaction system, which has a steady situation[1]. This means that the rate for each enabled transition in each marking is constant. The steady situation[1] concentrations and rates can be determined using simulation of the SPN. Examples for which about 10,0000 simulation runs have been done may be considered. These runs has to be merged into one averaged simulation run showing the mean of the concentrations, and thus also of the rates, over the time. We take the expectation values of the steady situation[1] rates for our investigations.

Simulation means approximation; thus it is not a priori clear how accurate the determined steady situation[1] rates are.

The reciprocal value of the rate is the time which each enabled transition has to wait before it can fire. The transition with the minimal waiting time fires in an SPN. Consequently, an SPN acts in the steady situation[1] exactly like a certain kind of TPN: we propose to construct a TPN, having the same underlying Petri net as the SPN, and where the transitions t will recieve time intervals $[a_t, b_t]$,

where $a_t = b_t$ is equal to the above calculated waiting time of t in the SPN in the steady-state.

A qualitative analysis of the TPN can prove whether the subset of all reachable states generate cycle(s) only (up to some initiation part). Furthermore, the time length of these cycles can be computed.

The formal analysis we proposed allows the following interpretations. If the reduced reachability graph of the TPN consists of cycles, then the considered rates achieved by simulation describe a steady situation[1], actually. By contraposition we may deduce, that a non cyclic form of the reduced state space indicates severe problems in the set of data used to build the TPN, and by consequence, in the initially established data from experiences in the wet labs.

Additionally the time-length of the cycles can be easily computed and compared with results from the wet labs. Both, the reachability graph of the TPN and the time-length of the cycles are either an indication for the correctness of the models or they invalidate these. Therefore our method offers a way of validating a complex modeling process of biochemical reaction systems.

4   An Example

In this section we will illustrate our approach of analysing a biochemical reaction system in a steady situation[1] along an example introduced in [4] and studied further in [6, 8], concerning the core model of the influence of the Raf-1 Kinase Inhibitor Protein (RKIP) on the Extracellular signal Regulated Kinase (ERK) signalling pathway.

In [4] this biochemical reaction system is modeled using an integrated approach of mathematical modeling in combination with experimental data. This model consists of eleven nonlinear ODEs. The parameters in the ODEs are estimated using interpolation of polynomial functions.

Afterwards, simulation studies provides a qualitative validation of the mathematical model compared to experimental results in the wet labs in view of the transient behavior and sensitivity analysis. However, parameter estimation is, as already mentioned, an uncertain factor in such a mathematical model.

Then in [6, 8], a qualitative model is proposed in terms of a Petri Net, see Fig. 3, deduced from the quoted ODE system.

Additionally, reaction rates are associated to all transitions of this PN [6, 8]. These are derived from the estimated parameters used in [4]. The obtained whole model is therefore an SPN. In [6], inter alia, the example is considered w.r.t. the attained steady situation[1] in the biochemical network. This is done by simulation: Rate values for the reactions are estimated after about 10.000 simulation runs have been done. Nevertheless, considering the behaviour of the model based on estimated parameters, we are in presence of a further factor of uncertainty.

We are going to investigate the SPN in the simulated steady situation[1]. First let us have a look on its reachabilty graph depicted on the left hand side of Fig.4. Unfortunately no method exists, to our knowledge, to find out analytically
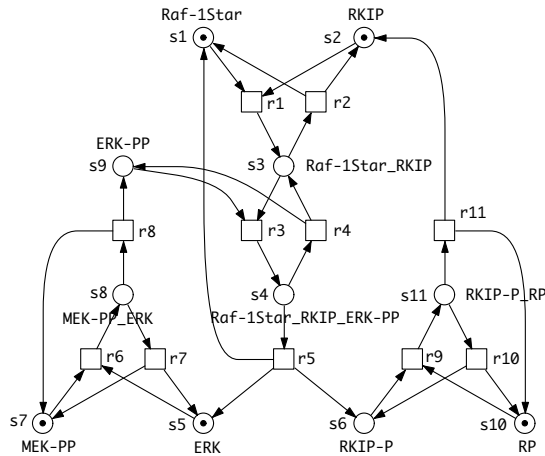
Fig. 3. The Petri net for the core model of the RKIP pathway, consisting of 11 places and 11 transitions. The places s1, ..., s11 stand for proteins or protein complexes. Complexes are indicated by an underscore _ between the protein names, phosphorylated forms by the suffix -P or -PP. The transitions r1, ..., r11 model the reactions. The preplaces of a transition correspond to the reaction's precursors, and its postplaces to the reaction's products. The layout follows the suggestions by the graphical notation used in [4]. The initial marking is constructed systematically using standard Petri net analysis techniques. *This figure with its legend is cited from [8].*

/ formally which are the states (nodes) corresponding to the steady situation[1]. Instead, we will use the estimated data in order to derive a TPN. This time-dependent Petri net should have the same state space as the SPN in the steady situation[1]. To be able to do this derivation, we need to know the waiting (or delay) times τ_i. They can be calculated from three kind of informations/data, given in the tables below.

- the **rate function** v_i, presented in [8] for each of the eleven transitions r_i, and shown in Table 1

- the estimated parameters $k_1 \cdots k_{11}$ in the eleven corresponding ODEs, presented in [4], renamed **rate parameters** and denoted by $c_i := k_i$ in [8], and shown in Table 1

- the **mean steady situation[1] concentrations** for the species s1 $\cdots$ s11 are taken from [6] and shown in Table 2.

Table 1. The rate function for each transition and the rate constants (the estimated parameter in the ODEs), *cited from [4, 8]*. The abbreviations s1 $\cdots$ s11 stand for the involved species as follows: s1 is Raf-1*, s2 is RKIP, s3 is Raf-1*_RKIP, s4 is Raf-1*_RKIP_ERK-PP, s5 is ERK, s6 is RKIP-P, s7 is MEK-PP, s8 is MEK-PP_ERK, s9 is ERK-PP, s10 is RP and s11 is RKIP-P_RP. In the rate functions each of the s1 $\cdots$ s11 is supposed to be the mean concentration of the species s1 $\cdots$ s11 in the simulated steady situation in the SPN, as given in Table 2.

transition r_i	rate function v_i	rate constant c_i
$r1$	$c1 \cdot s1 \cdot s2$	0.053
$r2$	$c2 \cdot s3$	0.0072
$r3$	$c3 \cdot s3 \cdot s9$	0.625
$r4$	$c4 \cdot s4$	0.00245
$r5$	$c5 \cdot s4$	0.0315
$r6$	$c6 \cdot s5 \cdot s7$	0.8
$r7$	$c7 \cdot s8$	0.0075
$r8$	$c8 \cdot s8$	0.071
$r9$	$c9 \cdot s6 \cdot s10$	0.92
$r10$	$c10 \cdot s11$	0.00122
$r11$	$c11 \cdot s11$	0.87

Table 2. Mean steady situation[1] concentrations for for all s_i, *cited from [6]*.

specie s_i	concentration
$s1$	0.2133
$s2$	0.1727
$s3$	0.2163
$s4$	0.5704
$s5$	0.0332
$s6$	0.0200
$s7$	0.7469
$s8$	0.2531
$s9$	0.1433
$s10$	0.9793
$s11$	0.0207

Now, we can calculate the rates $v_1 \cdots v_{11}$ of the transitions in the steady situation[1] using their rate functions from Table 1 and the data from Table 1 and Table 2. Subsequently, the delay time τ_i for every one of the ten transitions r_i is obtained as the reciprocal of the rate in the steady situation[1]. The resulting values are presented in Table 3.

Finally, the TPN model can be constructed: As skeleton of the TPN model we take the underlying PN of the SPN, i.e. the net given in Fig. 3. To each

Table 3. Rates in the steady situation and delay times for r1 $\cdots$ r11.

transition r_i	rate in the steady state v_i	delay time in the steady state τ_i (rounded)
$r1$	0.00195235623	512
$r2$	0.00155736	642
$r3$	0.019372369	52
$r4$	0.00139748	716
$r5$	0.0179676	56
$r6$	0.019837664	50
$r7$	0.00189825	527
$r8$	0.0179701	56
$r9$	0.01801912	55
$r10$	0.000025254	39598
$r11$	0.018009	56

transition r_i, $1 \leq i \leq 11$, a time interval $[\tau_i, \tau_i]$ is associated, where τ_i is the calculated delay time, from Table 3.
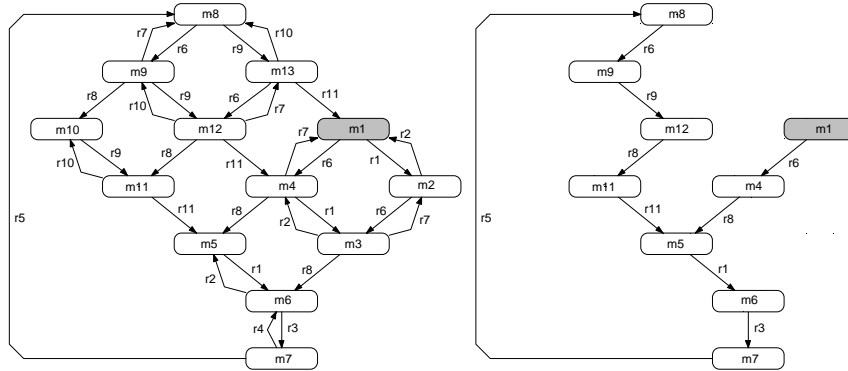

Fig. 4. Left hand: The reachability graph, from [8], for the SPN of Fig. 3.
Right hand: The reachable p-markings in the TPN. Please note, that this is not the reachability graph of the TPN.

Now the obtained TPN may be analysed. First, we just calculate the reachable p-markings, designed on the right hand side of Fig.4. We observe that due to the time constraints this graph has 9 nodes, i.e., much less p-markings are reachable as in the reachability graph of the SPN, depicted on the left hand side, which is also the reachability graph of the underlying net of the SPN and TPN. We also detect that the right graph is clearly a subgraph of the left one. The complete state space of the TPN is -a priori- infinite, a lot of states may share the same p-marking.

The reduced reachability graph of the considered TPN can now be constructed, by applying the reduction method described in section 2.3. We did it with tools INA and Charlie [7, 17], which gave us the same result, depicted in Fig 5. It consists of eleven essential states, i.e., 10 pairs of p- and t-markings, although only 10 p-markings are reachable in the considered TPN. This is no incoherence : Two essential states, $z3$ and $z10$, share the same p-marking $m5$. However, in the cycle each p-marking belongs to exactly one state (node) only.
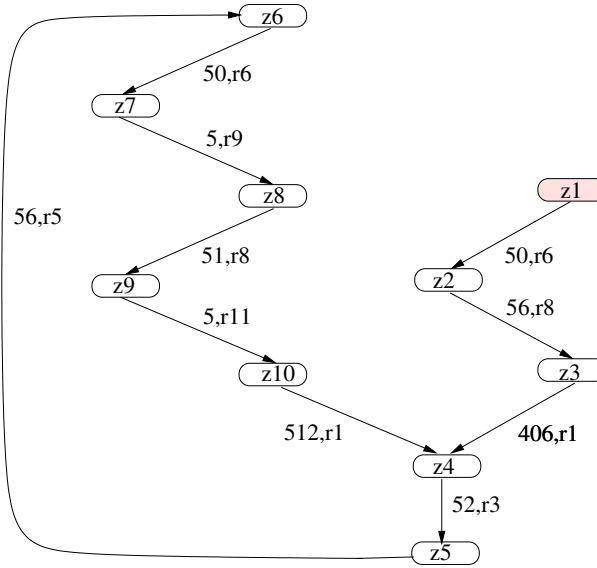

Fig. 5. The reduced reachability graph of the TPN model.

Analyzing this reduced reachability graph of the TPN tells us that it consists of the **cycle** $z4, r3, z5, r5, z6, r6, z7, r9, z8, r8, z9, r11, z10, r1$ and an **initiation path** $z1, r6, z2, r8, z3, r1$.

This path (or panhandle) is caused by the choice of the initial p-marking for the TPN, chosen to be the same as for the SPN. Actually, the initial p-marking for the TPN should be a p-marking which the SPN reaches in the steady situation[1]. However, the TPN only initiates its behaviour by this path and then comes to the steady situation[1], i.e. stays in the cycle. The time-length of the cycle was also calculated, its value is 731 time units.

We also read on this reduced graph that the transitions $r2, r4, r7$ and $r10$ will never fire. Such transition are called *dead*. These are the transitions modeling the backward reactions which have rate constants being essentially smaller as the

rate constants for the forward reactions. This tell us that in the steady situation[1] the backward reactions do never proceed.

5   Conclusions

In this paper we introduce a method for qualitative and quantitative evaluation of an SPN model of biochemical reaction systems in a steady situation[1] including validation of all used data. A mathematical model of such a system contains a number of uncertain factors resulting from the estimation of the parameters in the ODEs and the values of the reaction rates in the steady situation[1] obtained by simulation of the SPN. The reaction rates and the concentration of the species in the steady situation[1] are constant values.

This means that the set of reachable markings in the SPN model in the steady situation[1] is finite and they generate a cycle, not necessarily a simple one. But no state reachability analysis of the SPN does allow for isolating those states which correspond to the steady situation[1] i.e. does allow to detect the cycle.

Due to the fact of constant values, we are able to propose a mapping from the usual SPN model in the steady situation[1] onto a TPN model which has the same behaviour. Contrarily to the SPN model, we can reduce the state space of the TPN to the part we are interested in, consisting of the essential states. The obtained reduced state graph can be further analyzed. Its cyclic or non cyclic form validate or invalidate the used data during the modeling process. The time length of the cycle can be calculated and compared to real time measures, too.

The algorithms for reachability analysis of TPNs, implemented in the tools [3,7,17] had been applied for the evaluation of the simulated steady situation[1] in a mathematical model of our running example. We considered the core model of the influence of the Raf-1 Kinase Inhibitor Protein (RKIP) on the Extracellular signal Regulated Kinase (ERK) signalling pathway. We were able to show that the simulated values for the reaction rates define one cycle in the TPN model and to compute the time-length of this cycle. Furthermore we ascertain that the backward reactions do not proceed in the steady situation[1]

We will lead some reflexions if the initial state for the TPN could be redefined in a better way by regarding the values for the concentrations of the species in the simulated steady situation[1] We are planning to apply the presented method to some other cases of biological or biochemical interaction networks, were more complex steady situations[1] with -a priori- non simple cycles.

References

1. M. Ajmone Marsan. Stochastic Petri nets: An elementary introduction. In *In Advances in Petri Nets*, pages 1–29. Springer, 1989.
2. M. Ajmone Marsan, G. Balbo, G. Conte, S. Donatelli, and G. Franceschinis. *Modelling with Generalized Stochastic Petri Nets.* John Wiley and Sons, 1995.
3. B. Berthomieu. *TIme petri Net Analyzer.* LAAS / CNRS, 7, avenue du Colonel Roche, 31077 Toulouse, France, http://www.laas.fr/bernard/tina/, 2.9.8 released edition, 2009.

4. K.-H. Cho, S.-Y. Shin, H.-W. Kim, O. Wolkenhauer, B. McFerran, and W. Kolch. Mathematical modeling of the influence of RKIP on the ERK signaling pathway. *Lecture Notes in Computer Science*, 2602:127–141, 2003.

5. R. David and H. Alla. *Petri Nets and Grafcet: Tools for Modelling Discrete Event Systems*. Prentice Hall, New York London Toronto Sydney Tokyo Singapure, 1992.

6. D. Gilbert and M. Heiner. From Petri Nets to Differential Equations - an Integrative Approach for Biochemical Network Analysis. In *Proc. ICATPN 2006, Turku, June, Springer LNCS 4024*, pages 181–200, 2006.

7. M. Heiner. *Charlie*. Brandenburgische Technische Universität, http://www-dssz.informatik.tu-cottbus.de/DSSZ/Software/Charlie, January 2011.

8. M. Heiner, R. Donaldson, and D. Gilbert. *Petri Nets for Systems Biology*, chapter 3, pages 61–97. Jones & Bartlett Learning, LCC, 2010.

9. G. Koh, D. Hsu, and P.S. Thiagarajan. Incremental Signaling Pathway Modeling by Data Integration. In *Proc. of the 14th International Conference on Research in Computational Molecular Biology (RECOMB 2010), Lisabon, Portugal*, 2010.

10. H. Matsuno, Y. Tanaka, H. Aoshima, A. Doi, M. Matsui, and S. Miyano. Biopathways representation and simulation on hybrid functional Petri net. *Silico Biology*, 3(3):2592–2601, 2003.

11. L. Popova. On Time Petri Nets. *J. Inform. Process. Cybern. EIK 27(1991)4*, pages 227–244, 1991.

12. L. Popova-Zeugmann. Time Petri Nets State Space Reduction Using Dynamic Programming. *Journal of Control and Cybernetics*, 35(3):721–748, 2007.

13. L. Popova-Zeugmann. *Time and Petri Nets (in German)*. Habilitation Thesis, Humboldt Universität zu Berlin, 2007.

14. L. Popova-Zeugmann. Quantitative evaluation of time-dependent Petri nets and applications to biochemical networks. *Natural Computing, Springer Netherlands*, pages 1–27, 2010.

15. L. Popova-Zeugmann and D. Schlatter. Analyzing Path in Time Petri Nets. *Fundamenta Informaticae (FI) 37, IOS Press, Amsterdam*, pages 311–327, 1999.

16. S. Soliman and H. Heiner. A Unique Transformation from Ordinary Differential Equations to Reaction Networks. *PLoS ONE 5(12): e14284*, 2010.

17. H.-P. Starke. *INA – The Intergrated Net Analyser*. Humboldt Universität zu Berlin, http://www2.informatik.hu-berlin.de/∼ starke/ina.html, 2003.

18. A. Yartseva, R. Devillers, H. Klaudel, and F. Képès. From MIN model to ordinary differential equations. *J. Integrative Bioinformatics*, 4(3), 2007.