Sepinoud Azimi | Eugen Czeizler | Cristian Gratie | Diana Gratie | Bogdan Iancu | Nebiat Ibssa | Ion Petre | Vladimir Rogojin | Tolou Shadbahr | Fatemeh Shokri

An Excursion through Quantitative Model Refinement

Turku Centre *for* Computer Science

TUCS Technical Report
No 1150, November 2015

An Excursion through Quantitative Model Refinement

Sepinoud Azimi | Eugen Czeizler | Cristian Gratie
Åbo Akademi University, Department of Computer Science
Turku Centre for Computer Science (TUCS)
{sazimi, eczeizle, cgratie}@abo.fi

Diana Gratie | Bogdan Iancu
Åbo Akademi University, Department of Computer Science
Turku Centre for Computer Science (TUCS)
{dgratie, biancu}@abo.fi

Nebiat Ibssa
Department of Information Technology, University of Turku
nebiat.ibssa@abo.fi

Ion Petre | Vladimir Rogojin
Åbo Akademi University, Department of Computer Science
Turku Centre for Computer Science (TUCS)
{ipetre, vrogojin}@abo.fi

Tolou Shadbahr | Fatemeh Shokri
Åbo Akademi University, Department of Computer Science
{tshadbahr, fshokri}@abo.fi

TUCS Technical Report
No 1150, November 2015

Abstract

There is growing interest in creating large-scale computational models for biological process. One of the challenges in such a project is to fit and validate larger and larger models, a process that requires more high-quality experimental data and more computational effort as the size of the model grows. Quantitative model refinement is a recently proposed model construction technique addressing this challenge. It proposes to create a model in an iterative fashion by adding details to its species, and to fix the numerical setup in a way that guarantees to preserve the fit and validation of the model. In this survey we make an excursion through quantitative model refinement – this includes introducing the concept of quantitative model refinement for reaction-based models, for rule-based models, for Petri nets and for guarded command language models, and to illustrate it on three case studies (the heat shock response, the ErbB signaling pathway, and the self-assembly of intermediate filaments).

Keywords: Quantitative model refinement, heat shock response, intermediate filaments, Erbb signalling, acetylation.

TUCS Laboratory
Computational Biomodelling Laboratory (COMBIO)

1 Introduction

Building and analysing large-scale models has attracted much attention recently as shown, e.g., by building whole-cell models [24] or organ models [2, 34]. This is supported by advancement of biotechnologies, especially in terms of growing amounts of experimental data leading to a deeper understanding of the functions of a cell. On the other hand, the computational techniques for building biomodels have seen in contrast more modest progress. The most commonly used technique today is to compile a collection of submodels and to focus the computational effort on the communication and compatibility between them. This is a rather ad-hoc approach, highly sensitive to availability of existing submodels and vulnerable even to minor changes in them.

We discuss in this paper an approach for building large-scale models based on the idea of iteratively building the model through adding details to it step-by-step so that its experimental fit and validation is preserved in each step. This allows the modeler to start with an abstract view of the model and to add details to it as they become available; it also allows the modeler to deal with a hierarchy of models and to easily zoom-in and -out to various levels of detail as needed in various applications. Several methods have been proposed to facilitate model refinement in different frameworks, e.g., ODE-based models [19, 9], rule-based models [29], Petri nets [37], biochemical reaction networks [21], π-calculus [33].

This paper is thought of as an excursion through quantitative model refinement, introducing briefly the concept of fit-preserving refinement in several modeling frameworks and demonstrating it on three case-studies. It is only partially self-contained due to space restrictions; instead it indicates in many places references for further reading on each topic. The paper is structured as follows. In Section 2 we introduce reaction-based models and their associated ODE-based mass-action semantic. In Section 3 we introduce the main concept of this paper, that of quantitative model refinement; we also formulate a necessary and sufficient condition for how the numerical details of a refined model should be set so that it preserves the fit and the validation of the initial model. In Section 4 we introduce our three case studies: the heat shock response, the ErbB signaling pathway, and the self-assembly of intermediate filaments. In Section 5 we discuss two software implementations of the quantitative model refinement. In Section 6 we discuss the concept of model refinement in the context of rule-based, Petri nets, and guarded command language modeling. We conclude the paper with a short discussion in Section 7.

2 Preliminaries

We recall in this section some of the basic notions and definitions we need throughout the paper. For more details we refer to [20, 5, 9, 11].

2.1 Reaction-based models

In this section we briefly introduce the notion of *reaction-based models* following the notations in [9, 11].

A reaction-based model $N = (\mathscr{S}, \mathscr{R})$ consists of a set of species $\mathscr{S} = \{S_1, S_2, ..., S_m\}$ and a set of *reactions* $\mathscr{R} = \{r_1, \dots, r_n\}$. A *reaction* r_j is of the form:

$$r_j : \sum_{i=1}^{m} c_{ij} S_i \rightarrow \sum_{i=1}^{m} d_{ij} S_i,$$

where $c_{i,j}, d_{i,j} \in \mathbb{N}$, $1 \leq i \leq m$, $1 \leq j \leq n$. A reaction can also be described as:

$$r_j : \mathbf{c_j} \rightarrow \mathbf{d_j},$$

where $\mathbf{c_j} = (c_{1j}, c_{2j}, ..., c_{mj})^T$ and $\mathbf{d_j} = (d_{1j}, d_{2j}, ..., d_{mj})^T$ are called the left- and right-*complex* of reaction r_j, resp.

The *stoichiometric coefficient* of species S_i in reaction r_j is denoted by s_{ij} and defined as $s_{ij} = d_{ij} - c_{ij}$. We say a species S_i is *produced* in reaction r_j of N, if If $s_{ij} > 0$, and that it is *consumed* otherwise.

A mass-action reaction-based model is described as $M = (\mathscr{S}, \mathscr{R}, \mathbf{k})$ where $N = (\mathscr{S}, \mathscr{R})$ is a reaction-based model and $\mathbf{k} = (k_{r_1}, \dots, k_{r_n}) \in \mathbb{R}^{\mathscr{R}}_{\geq 0}$. We call $k_{c \rightarrow d}$ the reaction rate constant of reaction $\mathbf{c} \rightarrow \mathbf{d}$.

2.2 ODE-based mass-action model

We introduce here the *ODE-based mass-action model* corresponding to a reaction-based model; for details we refer to [25, 13]. In an ODE the dynamics of a system is expressed in terms of the time-dependent evolution of each species' concentration. We assume that the concentrations of the species is only affected by the reaction. In the case of an ODE model the time evolution of any S_i concentration can be considered as a function $s_i : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$. We define s_i in the case of mass-action kinetics, through the following system of ODEs:

$$\dot{s_i} = \sum_{j=1}^{n} (d_{ij} - c_{ij}) k_j \prod_{q=1}^{m} s_q^{c_{qj}},$$

where $\dot{s}$ denotes the differential of s. We define the system of ODE for all species in a compact form as:

$$\dot{\mathbf{s}} = \sum_{\mathbf{c} \rightarrow \mathbf{d} \in \mathscr{R}} k_{\mathbf{c} \rightarrow \mathbf{d}} \mathbf{s}^c (\mathbf{d} - \mathbf{c}),$$

where $\mathbf{s} = (s_1, s_2, ..., s_m)^T$, $\dot{\mathbf{s}} = (\dot{s_1}, \dot{s_2}, ..., \dot{s_m})^T$ and $\mathbf{s}^c = \prod_{i=1}^{m} s_i^{c_i}$.

Note that we only consider irreversible reactions since any reversible reaction in the form of

$$r_j : \sum_{i=1}^{m} c_{ij} S_i \rightleftarrows \sum_{i=1}^{m} d_{ij} S_i$$

can also be written as two different irreversible reactions:

$$r_j^{(1)} : \sum_{i=1}^{m} c_{ij} S_i \rightarrow \sum_{i=1}^{m} d_{ij} S_i, \quad r_j^{(2)} : \sum_{i=1}^{m} d_{ij} S_i \rightarrow \sum_{i=1}^{m} c_{ij} S_i.$$

3 Quantitative model refinement

The top-down development of large biological models starts with an initial abstraction of the considered biological phenomena, which can then iteratively extended by adding details to it. In the context of reaction-based models relying on mass-action kinetics, one can distinguish *data refinement*, which consists in the replacement of one (or more) species with several variants, i.e. *subspecies*, and *process refinement*, where a generic reaction is replaced with a set of reactions that captures the process in more details by providing intermediary steps. We focus here on data refinement.

Building models via refinement becomes increasingly difficult as the model size grows. Generating the refined reactions manually is both tedious and error prone. To address this, one can rely on *structural refinement*, which provides a generic and systematic approach for generating refined reactions based on the desired refinement of species. Furthermore, fitting a large model is a computationally expensive process and thus it becomes critical that, to the extent possible, the computational effort spent on fitting previous versions of the model is not completely wasted, but instead the obtained parameter values are reused for the initialization of the refined model. This can be accomplished via *fit-preserving refinement*, where the parameters of the refined model are set up so as to capture the same dynamics with respect to the species of the original model.

3.1 Structural refinement

In this subsection we discuss structural data refinement, as introduced in [9]. We start with the definition of species refinement, which aims to capture the replacement of species from the original model with subspecies in the refined one.

Definition 3.1 ([9]) *Let $\mathscr{S}$ and $\mathscr{S}'$ be two sets of species. A relation $\rho \subseteq \mathscr{S} \times \mathscr{S}'$ is a* species refinement relation *if and only if it satisfies the following conditions:*

1. *for each $S \in \mathscr{S}$, $\rho(S) \neq \varnothing$;*
2. *for each $S' \in \mathscr{S}'$ there exists exactly one $S \in \mathscr{S}$ such that $S' \in \rho(S)$.*

For each $S \in \mathscr{S}$ we denote $\rho(S) = \{S \in \mathscr{S}' \mid (S, S') \in \rho\}$. Intuitively, the constraints from the definition ensure that each species from the original model is refined to at least one subspecies (more than one in the case of non-trivial refinements) and each species of the refined model corresponds to exactly one "parent" species from the original model. A species refinement ρ can also be written as an $(\mathscr{S} \times \mathscr{S}')$-matrix with $\{0, 1\}$ entries, referred to as the *characteristic matrix* of ρ, defined as follows:

$$\boldsymbol{M}_\rho = (m_{S,S'})_{S \in \mathscr{S}, S' \in \mathscr{S}'}, \qquad m_{S,S'} = \begin{cases} 1, & \textit{if } S' \in \rho(S) \,; \\ 0, & \textit{otherwise} \,. \end{cases}$$

Note that each column of the matrix has exactly one 1-entry.

The species refinement relation induces the structural refinement of complexes, reactions and reaction networks.

Definition 3.2 ([11]) *Let $\mathscr{S} = \{S_1, \ldots, S_m\}$ and $\mathscr{S}' = \{S'_1, \ldots, S'_{m'}\}$ be two sets of species and $\rho \subseteq \mathscr{S} \times \mathscr{S}'$ a species refinement relation.*

1. *Let $\boldsymbol{c} = [c_1, \ldots, c_m]^T \in \mathbb{N}^{\mathscr{S}}$ and $\boldsymbol{c'} = [c'_1, \ldots, c'_{m'}] \in \mathbb{N}^{\mathscr{S}'}$ be two complexes over $\mathscr{S}$, respectively $\mathscr{S}'$. We say that $\boldsymbol{c}'$* is a ρ-refinement of $\boldsymbol{c}$*, denoted $\boldsymbol{c}' \in \rho(\boldsymbol{c})$, if*

$$\sum_{\substack{1 \le j \le m' \\ S'_j \in \rho(S_i)}} c'_j = c_i, \quad \textit{for all } 1 \le i \le m$$

   or, equivalently, if $\boldsymbol{c} = \boldsymbol{M}_\rho \boldsymbol{c'}$.

2. *Let $r : \boldsymbol{c} \to \boldsymbol{d}$ be a reaction over $\mathscr{S}$ and $r' : \boldsymbol{c'} \to \boldsymbol{d'}$ a reaction over $\mathscr{S}'$. We say that r'* is a ρ-refinement of r*, denoted $r' \in \rho(r)$, if $\boldsymbol{c'} \in \rho(\boldsymbol{c})$ and $\boldsymbol{d'} \in \rho(\boldsymbol{d})$.*

3. *Let $N = (\mathscr{S}, \mathscr{R})$ and $N' = (\mathscr{S}', \mathscr{R}')$ be two reaction-based models. We say that N'* is a ρ-refinement of N*, denoted $N' \in \rho(N)$, if*

$$\mathscr{R}' \subseteq \bigcup_{r \in \mathscr{R}} \rho(r) \quad \textit{and} \quad \rho(r) \cap \mathscr{R}' \neq \varnothing, \textit{ for all } r \in \mathscr{R}.$$

   In case $\mathscr{R}' = \bigcup_{r \in \mathscr{R}} \rho(r)$, we say that N' is the full ρ-refinement of N.

4. *Let $M = (\mathscr{S}, \mathscr{R}, \boldsymbol{k})$ and $M' = (\mathscr{S}', \mathscr{R}', \boldsymbol{k'})$ be two mass-action reaction-based models. We say that M'* is a ρ-refinement of M*, denoted $M' \in \rho(M)$, if $(\mathscr{S}', \mathscr{R}') \in \rho(\mathscr{S}, \mathscr{R})$. We say that M'* is a full ρ-refinement of M *if $(\mathscr{S}', \mathscr{R}')$ is the full ρ-refinement of $(\mathscr{S}, \mathscr{R})$.*

5. *Let $\boldsymbol{\sigma} \in \mathbb{R}^{\mathscr{S}}$ and $\boldsymbol{\sigma'} \in \mathbb{R}^{\mathscr{S}'}$ (thought of as the initial values for the system of ODEs associated to M and M'). We say that $\boldsymbol{\sigma'}$* is a ρ-refinement of $\boldsymbol{\sigma}$*, denoted $\boldsymbol{\sigma'} \in \rho(\boldsymbol{\sigma})$, if $\boldsymbol{\sigma} = \boldsymbol{M}_\rho \boldsymbol{\sigma'}$.*

Example 3.1 *Consider the reaction $A + B \xrightarrow{k} 2B$. We refine the reaction to include two different subtypes of species A, A_1 and A_2; species B remains unchanged but for the lack of clarity we denote it in the refined model by a new variable, say B_1. The corresponding species refinement relation is given by $\rho = \{(A, A_1), (A, A_2), (B, B_1)\}$. The two possible refinements of the considered reaction are $A_1 + B_1 \xrightarrow{k_1} 2B_1$, $A_2 + B_1 \xrightarrow{k_2} 2B_1$.*

Note that this reaction is part of the Lotka-Volterra model; for a complete discussion of the refinement of this model we refer to [5, 10].

3.2 Fit-preserving refinement

In this subsection we define the fit-preserving refinement, as introduced in [5], with the notations and formal definition from [9]. Given an initial value problem, i.e. an ODE $\dot{\boldsymbol{x}} = \boldsymbol{F}(\boldsymbol{x})$ with the initial condition $\boldsymbol{x}(0) = \boldsymbol{x_0}$, we use $\boldsymbol{x}[\boldsymbol{x_0}]$ to denote its (unique) solution.

The problem we investigate in this section is the following:

What is the numerical setup (kinetic rate constants and initial values) of a refined model ensuring that for each species of the basic model, its corresponding function in the mathematical model is the sum of the functions corresponding to its subspecies in the refined model?

The problem is strongly motivated by the need to preserve the numerical fit of an already validated model, while allowing its extension with additional details through quantitative model refinements. We give this problem a solution in this section and we use this solution in several different frameworks and case-studies in the remaining of the paper.

Definition 3.3 *Let $M = (\mathscr{S}, \mathscr{R}, \boldsymbol{k})$ and $M' = (\mathscr{S}', \mathscr{R}', \boldsymbol{k'})$ be two mass-action reaction networks and $\rho \subseteq \mathscr{S} \times \mathscr{S}'$ a species refinement relation. For any $\boldsymbol{\sigma} \in \mathbb{R}^{\mathscr{S}}_{\geq 0}$ and $\boldsymbol{\sigma'} \in \mathbb{R}^{\mathscr{S}'}_{\geq 0}$ we denote by $\boldsymbol{s}[\boldsymbol{\sigma}] : [0, \tau) \to \mathbb{R}^{\mathscr{S}}_{\geq 0}$ ($\boldsymbol{s'}[\boldsymbol{\sigma'}] : [0, \tau') \to \mathbb{R}^{\mathscr{S}'}_{\geq 0}$) the vector of the real functions obtained as solutions of the ODE system associated to M (to M', resp.) with initial values $\boldsymbol{\sigma}$ ($\boldsymbol{\sigma'}$, resp.).*

We say that M' is a ρ-fit-preserving refinement of M *if $M' \in \rho(M)$ and, for all $\boldsymbol{\sigma} \in \mathbb{R}^{\mathscr{S}}_{\geq 0}$ and $\boldsymbol{\sigma'} \in \mathbb{R}^{\mathscr{S}'}_{\geq 0}$ such that $\boldsymbol{\sigma} = \boldsymbol{M}_\rho \boldsymbol{\sigma'}$, we have that*

$$\boldsymbol{s}[\boldsymbol{\sigma}](t) = \boldsymbol{M}_\rho \boldsymbol{s'}[\boldsymbol{\sigma'}](t),$$

for all values of t in a suitable right-neighborhood of 0.

Note that, for the same set of reactions, it is sometimes possible that two different assignments of kinetic rate constants lead to exactly the same ODE. For such models it is shown in [4] that the values of the rate constants can not be computed even from exact and complete experimental data for the system's dynamics.

As such, the requirement that a model has uniquely identifiable rate constants will be regarded as reasonable and desirable even outside the refinement framework.

What we are looking for is an effective procedure for assigning the values of the kinetic rate constants of the refined model so that a fit-preserving refinement is obtained. An (implicit) assignment that achieves this is given in Definition 3.4.

Definition 3.4 *Let $M = (\mathscr{S}, \mathscr{R}, \boldsymbol{k})$ and $M' = (\mathscr{S}', \mathscr{R}', \boldsymbol{k'})$ be two mass-action reaction networks and $\rho \subseteq \mathscr{S} \times \mathscr{S}'$ a species refinement relation. We say that M' is a* canonical ρ-refinement *of M if M' is a full ρ-refinement of M and, for every $\boldsymbol{c} \to \boldsymbol{d} \in \mathscr{R}$ and every $\boldsymbol{c'} \in \rho(\boldsymbol{c})$, we have that*

$$\sum_{\boldsymbol{d'} \in \rho(\boldsymbol{d})} k'_{\boldsymbol{c'} \to \boldsymbol{d'}} = \binom{\boldsymbol{c}}{\boldsymbol{c'}} k_{\boldsymbol{c} \to \boldsymbol{d}}\,, \textit{ where } \binom{\boldsymbol{c}}{\boldsymbol{c'}} = \frac{\prod_{i=1}^{|\mathscr{S}|} c_i!}{\prod_{j=1}^{|\mathscr{S}'|} c'_j!}\,.$$

It is shown in [9] that any canonical ρ-refinement is also a fit-preserving refinement. We provide here the stronger result of [11].

Theorem 3.1 ([11]) *Let $M = (\mathscr{S}, \mathscr{R}, \boldsymbol{k})$ and $M' = (\mathscr{S}', \mathscr{R}', \boldsymbol{k'})$ be two reaction networks such that M' is a full ρ-refinement of M.*

1. *If M' is a canonical ρ-refinement of M, then M' is a fit-preserving ρ-refinement of M.*

2. *If M has uniquely identifiable rate constants, then M' is a fit-preserving ρ-refinement of M if and only if M' is a canonical ρ-refinement of M.*

Note that Theorem 3.1 provides a complete characterization of fit-preserving refinement in the context of mass-action models. What is remarkable in this characterization is the linear dependency between the rate constants of the refined model and those of the original model.

Example 3.2 *Consider again the reaction from Example 3.1 and its refinements. In this case, canonical refinement translates to having $k_1 = k$ and $k_2 = k$, since the left hand sides of the two refined reactions are distinct.*

For a more comprehensive discussion of fit-preserving refinement, see [11], where several distinct fit-preserving refinements of the Brusselator [30] are presented and compared.

3.3 Refinement induced by the composition of species

In this subsection we rely on the initial refinement ideas proposed in [5], where a distinction is made between complex species (which consist of several, smaller, units, e.g. molecules composed of atoms) and atomic species, which can not be divided into smaller parts, within the current resolution of the model. For example, consider the following chemical reaction: $A + B{:}C \xrightarrow{k} A{:}B + C$.

The definition of refinement presented in Section 3 does not consider the composition of species. However, this information may be relevant, particularly in cases when the subspecies distinguished in the refined model are in fact induced by the data refinement of one (or several) atomic species. For the chosen reaction, note that there are three atomic species, namely A, B and C, and two complex species, $A{:}B$ and $B{:}C$. For uniformity, we assume that the reactants and products of a reaction are all complex species, thus we allow a complex species to be composed of a single atomic species.

Assume that in the refined model we can distinguish two types of B. We can write this as an *atomic refinement relation*:

$$\rho_{\text{atomic}} = \{(A, A_1), (B, B_1), (B, B_2), (C, C_1)\}.$$

This induces a refinement of all complex species of the model where, just as in the case of reaction refinement, we aim to capture all possible combinations of subspecies which are meaningful with respect to the composition of the species from the original model. In this case, the species refinement relation for complex species becomes

$$\rho = \{(A, A_1), (B{:}C, B_1{:}C_1), (B{:}C, B_2{:}C_1), (A{:}B, A_1{:}B_1), (A{:}B, A_1{:}B_2), (C, C_1)\}.$$

Given the species refinement relation ρ, structural refinement can proceed as in Section 3.1. The advantage of defining an atomic refinement is its compactness. Moreover, as we show in Section 5, this is enough for enabling the automated computation of the structural refinement of a model.

4 Case-studies

We introduce in this section the three case studies discussed in this paper: the heat shock response, the ErbB signaling pathway, and the self-assembly of intermediate filaments.

4.1 The heat shock response

The eukaryotic heat shock response is a conserved regulatory network that acts as a defence mechanism against proteotoxicity arising from environmental stressors such as: elevated temperature, toxins, infections, etc. Elevated temperatures induce protein misfolding leading to the formation of aggregates which hinder protein homeostasis, eventually bringing about apoptosis. The deleterious effects of elevated temperature upon proteins are counterbalanced by a family of molecular chaperones, called heat shock proteins, which bind to misfolded proteins, facilitating their recovery process so as to prevent apoptosis. We consider the following basic molecular model for the heat shock response, introduced in [32].

Heat shock proteins (hsp's) play a key role in the process of protein refolding, chaperoning misfolded proteins in the recovery process and facilitating the degradation of severely damaged proteins. Heat shock proteins possess an affinity towards misfolded proteins and, hence, they sequester them, form $\mathsf{hsp}\colon\mathsf{mfp}$ complexes, helping them recover to their original conformation (prot). The hsp-encoding genes transactivation controls the cell's response to environmental stressors. Gene transcription is regulated by a family of proteins, called heat shock factors (hsf's). Heat shock factors are found predominantly in the cell in a monomeric state when the cell does not withstand any stress from the environment, extensively bound to heat shock proteins ($\mathsf{hsp}\colon\mathsf{hsf}$). Elevated temperatures lead to the breakage of $\mathsf{hsp}\colon\mathsf{hsf}$, causing the release of hsf's. Heat stress induces the dimerization of heat shock factors (hsf_2) and their consequent trimerization (hsf_3), bringing them to a conformation which enables their binding with the promoter elements of the hsp-encoding gene, heat shock element(hse). This promotes hsp synthesis. However, once the expression level of hsp is elevated enough for the cell to endure the effects of environmental stressors, hsp synthesis is turned off. Heat shock proteins, thus, sequestrate the free hsf's, break dimers and trimers and impel DNA unbinding, by the formation of $\mathsf{hsp}\colon\mathsf{hsf}$ complexes. Consequently, the production of trimers is impeded. Temperature elevation causes proteins to misfold, as a consequence heat shock proteins are detached from heat shock factors, $\mathsf{hsp}\colon\mathsf{hsf}$ complexes being broken. Now free hsf's dimerize and trimerize, thus promoting the synthesis of hsp's. We list the complete set of reactions in Table 1.

Table 1: The molecular model for the eukaryotic heat shock response proposed in [32].

Reaction	Description
$2\,\mathsf{hsf} \rightleftarrows \mathsf{hsf}_2$	Dimerization (1)
$\mathsf{hsf} + \mathsf{hsf}_2 \rightleftarrows \mathsf{hsf}_3$	Trimerization (2)
$\mathsf{hsf}_3 + \mathsf{hse} \rightleftarrows \mathsf{hsf}_3\colon\mathsf{hse}$	DNA binding (3)
$\mathsf{hsf}_3\colon\mathsf{hse} \rightarrow \mathsf{hsf}_3\colon\mathsf{hse} + \mathsf{hsp}$	hsp synthesis (4)
$\mathsf{hsp} + \mathsf{hsf} \rightleftarrows \mathsf{hsp}\colon\mathsf{hsf}$	hsf sequestration (5)
$\mathsf{hsp} + \mathsf{hsf}_2 \rightarrow \mathsf{hsp}\colon\mathsf{hsf} + \mathsf{hsf}$	Dimer dissipation (6)
$\mathsf{hsp} + \mathsf{hsf}_3 \rightarrow \mathsf{hsp}\colon\mathsf{hsf} + 2\,\mathsf{hsf}$	Trimer dissipation (7)
$\mathsf{hsp} + \mathsf{hsf}_3\colon\mathsf{hse} \rightarrow \mathsf{hsp}\colon\mathsf{hsf} + 2\,\mathsf{hsf} + \mathsf{hse}$	DNA unbinding (8)
$\mathsf{hsp} \rightarrow \emptyset$	hsp degradation (9)
$\mathsf{prot} \rightarrow \mathsf{mfp}$	Protein misfolding (10)
$\mathsf{hsp} + \mathsf{mfp} \rightleftarrows \mathsf{hsp}\colon\mathsf{mfp}$	mfp sequestration (11)
$\mathsf{hsp}\colon\mathsf{mfp} \rightarrow \mathsf{hsp} + \mathsf{prot}$	Protein refolding (12)

Various post-translational modifications can affect heat shock factors (phosphorylation, acetylation, sumoylation) and influence DNA-binding activity. The heat shock response is attenuated as a result of the acetylation of heat shock factors (hsf's). We introduce here the refinement of hsf molecules as shown in [19], by considering the acetylation status of the hsf molecule at its $K80$ residue.

The species in the refined model are classified in two categories: *atomic* or *complex*. Atomic species refer to *self-contained* species, autonomous in their structure, see [12]. The structure of a complex however consists in at least two atomic species bound together. All species to be refined, previously mentioned above are atomic.

The refined model includes two types of heat shock factors: one to represent the acetylation of the lysine residue ($K80$) of hsf's and one for the non-acetylated hsf's. As a consequence, the $\mathsf{hsf_3}\colon\mathsf{hse}$ complex, for example, is to be refined into 4 subtypes conforming to the status of its every hsf molecule, considering the symmetry in the acetylation sites distribution: $\mathsf{rhsf_3}\colon\mathsf{rhse}$, $\mathsf{rhsf_3}^{(1)}\colon\mathsf{rhse}$, $\mathsf{rhsf_3}^{(2)}\colon\mathsf{rhse}$, $\mathsf{rhsf_3}^{(3)}\colon\mathsf{rhse}$. We denote by $\mathsf{rhsf_3}^{(\mathsf{i})}$: rhse the complex where i of the 3 hsf's are acetylated at site $K80$.

The refinement described above can be formalized through the species refinement relation below (one row for each species of the basic model):

$$\begin{aligned}\rho = \{&(\mathsf{hse},\mathsf{rhse}), (\mathsf{hsp},\mathsf{rhsp}), (\mathsf{prot},\mathsf{rprot}), (\mathsf{mfp},\mathsf{rmfp}), (\mathsf{hsp}\colon\mathsf{mfp},\mathsf{rhsp}\colon\mathsf{rmfp}),\\ &(\mathsf{hsf},\mathsf{rhsf}), (\mathsf{hsf},\mathsf{rhsf}^{(1)}),\\ &(\mathsf{hsf_2},\mathsf{rhsf_2}), (\mathsf{hsf_2},\mathsf{rhsf_2}^{(1)}), (\mathsf{hsf_2},\mathsf{rhsf_2}^{(2)}),\\ &(\mathsf{hsf_3},\mathsf{rhsf_3}), (\mathsf{hsf_3},\mathsf{rhsf_3}^{(1)}), (\mathsf{hsf_3},\mathsf{rhsf_3}^{(2)}), (\mathsf{hsf_3},\mathsf{rhsf_3}^{(3)}),\\ &(\mathsf{hsp}\colon\mathsf{hsf},\mathsf{hsp}\colon\mathsf{rhsf}), (\mathsf{hsp}\colon\mathsf{hsf},\mathsf{rhsp}\colon\mathsf{rhsf}^{(1)}),\\ &(\mathsf{hsf_3}\colon\mathsf{hse},\mathsf{rhsf_3}\colon\mathsf{rhse}), (\mathsf{hsf_3}\colon\mathsf{hse},\mathsf{rhsf_3}^{(1)}\colon\mathsf{rhse}), (\mathsf{hsf_3}\colon\mathsf{hse},\mathsf{rhsf_3}^{(2)}\colon\mathsf{rhse}),\\ &(\mathsf{hsf_3}\colon\mathsf{hse},\mathsf{rhsf_3}^{(3)}\colon\mathsf{rhse})\}.\end{aligned}$$

The refined model in [19] comprises 20 reactants and 55 irreversible reactions, while the initial model in [32] consists of 10 reactants and 17 irreversible reactions. The numerical details of the refined model, set in accordance with Theorem 3.1, can be found in [19]. Through refinement, the model preserves its fit and validation, even though its size increases considerably, both in number of reactants, and in number of reactions.

4.2 The ErbB signalling pathway

The ErbB signalling pathway is an evolutionary regulatory pathway, which plays a key role in the regulation of diverse cellular processes (growth, differentiation, motility, etc.) and whose anomalous behaviour is associated with cancer development in humans. The ErbB signalling pathway involves a number of cellular ligands, among which we are interested in this survey in EGF and HRG, and four receptor tyrosine kinases: ErbB1, ErbB2, ErbB3, ErbB4.

4.2.1 The initial ErbB signalling pathway model

The activation of the pathway commences with the binding of the epidermal growth factor (EGF) to the epidermal growth factor receptor EGFR (ErbB1), which brings about a dimerization of the newly formed complex and subsequently a rapid auto-phosphorylation of its tyrosine residues. The signal is propagated through two distinct pathways: Shc-dependent and Shc-independent, both of which lead to the activation of Ras-GTP. The Shc-dependent pathway is activated by the Shc protein, which binds to the dimerized, phosphorylated, ligand-bound receptor and then subsequently to Grb2.The Shc-independent pathway is in turn activated by the direct binding with Grb2. Both the aforementioned pathways require Sos to be recruited to the membrane. The pathway sustains an elaborate internalization process along with the degradation of several complexes. However, the recruitment of Sos impels an association with protein Ras which causes the activation of Ras in a GTP-dependent manner. Subsequent to its formation and activation, the inactivation of Ras-GTP is a consequence of the dissociation from the receptor complex involving protein GAP. It is not clear so far however what is the responsible kinase for the phosphorylation of Raf, but the model in [17] considers protein Raf to be phosphorylated by free Ras-GTP. Then in turn, subsequent to its phosphorylation, Raf is able to phosphorylate MEK. Doubly phosphorylated MEK sucessively phosphorylates ERK, see [17]. The initial model in [17] acknowledges the negative feedback loop from doubly phosporylated ERK to Sos, promoting as a result, the undbinding between Grb2-Sos and the receptor complex. Without any stimulation from EGF, the system is in a steady-state. The initial model described in [17] distinguishes between two pools of dually phosphorylated ERK (ERK-PP), first of which is identified in the cytoplasm and the latter in association to the internalized receptor. As described in [17], the model consists of 13 chemical processes: the activation of EGFR , the recruitment of the following proteins: Shc, Grb2 and Sos, the activation and the inactivation of Ras, the activation of Raf, the MEK phosphorylation/dephosphorylation, the dephosphorylation of ERK , the negative feedback from ERK to Sos, the internalization of receptor complexes and degradations reaction. A more elaborate discussion about the model can be found in [17]. The model has 103 species and 148 reactions.

4.2.2 The refined ErbB signaling pathway model

This subsection briefly describes the expansion of the EGFR signalling pathway model from [17] by means of fit-preserving data refinement, taking into account four members of the ErbB family: ErbB1 (EGFR), ErbB2 (HER2), ErbB3, ErbB4, and two ligands: EGF and HRG. The resulting model has 421 species and 928

reactions.

We consider only the following two refinements of two atomic species:

$$\mathsf{EGFR} \to \{\mathsf{ErbB1}, \mathsf{ErbB2}, \mathsf{ErbB3}, \mathsf{ErbB4}\}; \quad \mathsf{EGF} \to \{\mathsf{EGF}, \mathsf{HRG}\}.$$

Obviously, these refinements cascade to other refinements of complex species. We discuss this in the following.

Consider first the receptor activation reaction:

$$\mathsf{EGF} + \mathsf{EGFR} \overset{k_{lb}^{+}}{\leftrightarrow} \mathsf{EGF\text{-}EGFR}\,. \tag{1}$$

We refine it to include both ligands $L_1, L_2 \in \{\mathsf{EGF}, \mathsf{HRG}\}$ and the receptors $R_1, \ldots, R_4 \in \{\mathsf{ErbB1}, \mathsf{ErbB2}, \mathsf{ErbB3}, \mathsf{ErbB4}\}$ as follows:

$$L_i + R_j \xleftrightarrow[k_{i,j}^{-}]{k_{i,j}^{+}} L_i\text{–}R_j, \text{ for all } i = \overline{1,2}, j = \overline{1,4}.$$

We aim to set the kinetic rate constants of the refined model in concordance to the sufficient conditions for fit-preserving refinement in Section 3. Let's consider the ligand-binding reaction (1); its corresponding kinetic rate constants are set as follows: $k_{i,j}^{-} = k_{lb}^{-}$ and $k_{i,j}^{+} = k_{lb}^{+}$, for all $i = \overline{1,2}$, $j = \overline{1,4}$.

Consider now the dimerization of the ligand-bound receptor reaction:

$$2\,\mathsf{EGF\text{-}EGFR} \xleftrightarrow[k_{d}^{-}]{k_{d}^{+}} (\mathsf{EGF\text{-}EGFR})^2\,.$$

In the refined model we considered all possible combinations of ligand-bound receptor monomers, found on the *left-hand side* of the dimerization reactions. Since we have two types of ligands and four types of receptors, this gives us eight types of combinations ligand-receptors. Accordingly, the dimerization of the ligand-bound receptor is refined in the following manner:

$$C_i + C_j \xleftrightarrow[k_{i,j,l}^{-}]{k_{i,j,l}^{+}} (C_l)^2,$$

where $C_i, C_j, C_l \in \{\mathsf{EGF} - \mathsf{ErbBp}, \mathsf{HRG} - \mathsf{ErbBq}\,|p, q \in \overline{1,4}\}$. Note that we only consider the formation of homo-dimers in our considerations; hetero-dimers may also be included, with the consequence of drastically increasing the model size.

According to Theorem 3.1, the kinetic rate constants of the refined dimerization reaction are set as follows:

$$k_{i,j,l}^{+} = \begin{cases} 0, & \text{if } l \neq i; j \\ k_d^{+}, & \text{otherwise.} \end{cases} \quad ; \quad k_{i,j,l}^{-} = \begin{cases} 0, & \text{if } l \neq i; j \\ \frac{k_d^{-}}{8}, & \text{otherwise.} \end{cases}$$

Consider now the receptor production $\xrightarrow{k_p} \mathsf{EGFR}$, refined as $\xrightarrow{k_i} R_i$, $i = \overline{1,4}$, where $R_i \in$ {ErbB1, ErbB2, ErbB3, ErbB4}. The corresponding kinetic rate constants are set so as to comply with the condition in Theorem 3.1: $k_i = \frac{k_p}{4}$.

Finally, complex species of the initial model of [17] were refined taking into account all combinations of receptor-ligand binding. Let's take, for instance, a species $(\mathsf{EGF\text{-}EGFR}^*)^2\text{-}\mathsf{AC}$, where AC represents a so-called *chain of bound atomic species* (such as GAP-Grb2-Sos-Ras-GDP-Prot). According to our method, species $(\mathsf{EGF\text{-}EGFR}^*)^2\text{-}\mathsf{AC}$ was refined into the subspecies below:

$$(\mathsf{EGF\text{-}EGFR}^*)^2\text{-}\mathsf{AC} \to \{(C_i{}^*)^2 - AC\},\ \ 1 \leq i \leq 8,$$

with $C_i \in \{\mathsf{EGF} - \mathsf{ErbBp}, \mathsf{HRG} - \mathsf{ErbBq}\,|p, q = \overline{1,4}\}$ and with "$*$" character denoting the phosphorylation status the molecule.

4.3 Intermediate filaments self-assembly

Intermediate filaments (IF), together with actin filaments and microtubules, are the three types of protein filaments forming the cytoskeleton of eukaryotic cells [35]. IF in particular have an important role in the structural reinforcement of the cells and their organization into tissues, and in distributing the tensile forces across cells within a tissue [27]. IF sub-units are α-helical rods which assemble both laterally and using end-to-end interactions into rope-like filaments [15]. The emerging filaments range in length from hundreds of nm to micro-meter values, while their width (when in mature state) is preserved at 11 nm.

In the following we choose vimentin filaments as a representative for the class of intermediate filaments proteins, and we analyze their in-vitro assembly principles. Based on the recent studies in [6] and [28] we present both a well validated molecular and computational model of the in vitro vimentin assembly into filaments, as well as a refined model distinguishing between the different lengths of the emerging filaments.

The in-vitro vimentin assembly process follows four stages. In the first stage, monomers associate laterally into dimers and then into tetrameres (denoted as T). The tetramer sub-units are the first chemically stable compounds in the IF assembly process, and, moreover, the assembly can be blocked/freezed before continuing further. This is why when modelling the in-vitro IF assembly this first stage is omitted, and the IF assembly is assumed to be starting from tetramer level. The second assembly phase consist of a series of further lateral associations: two tetrameres merge into an octamer (O), two octamers merge into a hexadecamer (H), and two hexadecamer merge into a unit length filament (ULF). ULFs (denoted as U) are the basic units of the emerging filament structures. In the third assembly phase the filaments start forming and elongating, by sub-sequent end-to-end associations of both ULFs and of shorter filaments. In the final assembly phase the filaments undergo a radial compaction, from an ULF diameter of about 15 nm to a filament diameter of about 11 nm [15]. Since within this last assembly phase the ULF per filament ratio does not suffer any further modifications, this stage does not bring any changes within the molecular model itself.

Depending on the number n of constituent ULF's within one filament, we can differentiate between the emergent assemblies based on their "size" n. A common problem in modelling self-assembly systems is dealing with the combinatorial explosion of all possible emergent assemblies as possible different species. In case of the IF model above, this translates into the problem of representing and reasoning about all the emergent filaments of size 1, 2, 3, etc. In [6], the authors introduce a well validated molecular and numerical model for in-vitro vimentin assembly. Within this model, see Table 2 a), the emerging filaments consisting of at least two ULFs are treated in a homogenous manner, and are captured within the same generic species F. With this assumption in place, the authors succeed to validate several experimental data sets on the time dependent mean length of the emerging vimentin IFs. The model however is not able to capture the time distribution of a particular length filament, say the time distribution of filaments containing exactly 3 ULFs.

Using the refinement method described in Section 3 we can refine the generic filament species F according to any desired (finite) resolution level. For example, for introducing a model distinguishing between all filaments of lengths 1 to 5 ULFs, as well as filaments containing at least 6 ULFs, we can use the species refinement relation below; the entire refined molecular model is described in Table 2 b):

$$\rho = \{(\mathsf{T}, \mathsf{rT}), (\mathsf{O}, \mathsf{rO}), (\mathsf{H}, \mathsf{rH}), (\mathsf{U}, \mathsf{rF}_1), (\mathsf{F}, \mathsf{rF}_2), (\mathsf{F}, \mathsf{rF}_3),\\ (\mathsf{F}, \mathsf{rF}_4), (\mathsf{F}, \mathsf{rF}_5), (\mathsf{F}, \mathsf{rF}_{\geq 6})\}.$$

Moreover, by setting the kinetic rate constants of the refined model as in Theorem 3.1, we can ensure that the newly generated refined model is preserving its predictions for mean filament length. This implies that the refined models is indeed also validating the experimental data sets used in [6]. The kinetic rate constants of the refined model may be chosen as described in Table 2 b).

5 Software support

We discuss in this section two software tools for implementing quantitative model refinement in practice.

5.1 ModelRef

We have developed a software tool called *ModelRef* [22] implementing fit-preserving model refinement for atomic-only species as described in Section 3. The user provides as the input a numerical model as well as the refinement criteria. The numerical model in the input contains a set of chemical species, their initial concentrations, set of chemical reactions and their reaction kinetic rates. In the refinement criteria one indicates the correspondence between original and the refined species.

a) Basic model		b) Refined model	
Reaction	Rate constant	Reaction	Rate constant
$T + T \rightarrow O$	k_t	$rT + rT \rightarrow rO$	$k_t{}' = k_t$
$O + O \rightarrow H$	k_o	$rO + rO \rightarrow rH$	$k_o{}' = k_o$
$H + H \rightarrow U$	k_h	$rH + rH \rightarrow rF_1$	$k_h{}' = k_h$
$U + U \rightarrow F$	k_u	$rF_1 + rF_1 \rightarrow rF_2$	$k_{(1,1)}{}' = k_u$
$U + F \rightarrow F$	k_{uf}	$rF_1 + rF_i \rightarrow rF_{i+1}, 1 \leq i \leq 4$	$k_{(1,i)}{}' = k_{uf}$
$F + F \rightarrow F$	k_{ff}	$rF_1 + rF_j \rightarrow rF_{\geq 6}, j \in \{5, \geq 6\}$	$k_{(1,j)}{}' = k_{uf}$
		$rF_2 + rF_2 \rightarrow rF_4$	$k_{(2,2)}{}' = k_{ff}$
		$rF_i + rF_i \rightarrow rF_{\geq 6},$ $,i \in \{3, 4, 5, \geq 6\}$	$k_{(i,i)}{}' = k_{ff}$
		$rF_2 + rF_3 \rightarrow rF_5$	$k_{(2,3)}{}' = 2\,k_{ff}$
		$rF_i + rF_j \rightarrow rF_{\geq 6},$ $, 2 \leq i < j \leq 5, i + j \geq 6$	$k_{(i,j)}{}' = 2\,k_{ff}$
		$rF_i + rF_{\geq 6} \rightarrow rF_{\geq 6}\ 2 \leq i \leq 5$	$k_{(i,\geq 6)}{}' = 2\,k_{ff}$

Table 2: The molecular models of the basic (a) and the refined (b) representations of the IF assembly process.

ModelRef generates the refined model as follows:

- Every species from the original model that should be refined is substituted with the corresponding set of the refinement species.
- Every reaction from the original model that includes species being refined either as reactants or products is substituted with the set of reactions including the respective refinement species. The resulting set of reactions and their kinetic rates are calculated as defined in Section 3.

ModelRef handles models in both *SBML* and *CPS* file formats. The *Systems Biology Markup Language (SBML)* is one of the most wide-spread open interchange formats for computer models of biological processes [18]. *CPS* is a native file format of *Complex Simulator Pathway (COPASI)* [16] for storing and exchanging biological models.

The refinement criteria should be provided in *CSV (Comma Separated Values)* table, where the first column contains names of the original biochemical species, while the right column contains a set of species that should substitute/refine the respective original species from the left column.

ModelRef is implemented as a Java library and it is deployed as a stand-alone Java console application, as an Anduril [31] component and as a web-based service. Anduril is an open source component-based workflow framework for scientific data analysis developed at the Computational Systems Biology Laboratory, University of Helsinki. Anduril provides and API that allows to integrate rapidly various existing software tools and algorithms into a single data analysis pipeline. An Anduril pipeline comprises a set of interconnected executable programs (called components) with well-defined I/O ports, where an output port of a component may be connected to the input ports of some number of other components.

The web-service allows for a user to upload on our web-server a numerical model in either *SBML* or *CPS* formats, the refinement criteria as *CSV* table, and then, it sends back to the user the resulting refined model in either *SBML* or *CPS* format.

Since *ModelRef* is implemented as a Java class library, its functionality can be extended by other developers and it can be directly incorporated into other Java programs. As an Anduril component, *ModelRef* can be easily incorporated into data analysis pipelines.

5.2 StructRef

StructRef relies on the data refinement induced by an atomic refinement relation, as described in Section 3.3. The software is thought of as an interactive tool allowing the modeler to specify the initial atomic refinement relation, but also to intervene and alter intermediary results to better fit prior knowledge about the model that is refined.

The software takes as input a model represented in the SBML format. Intermediary results are saved as XML files. The final output, the structural refinement of the input model, is represented as SBML, with the intermediary results inserted as annotations, to allow for their reuse.

The software works as follows:

- Species are read from the input model and their composition is inferred from their names. Currently, the software assumes that the atomic components of a complex species are separated by colons, but this can easily be extended in future versions. Moreover, at the end of this step, an XML is generated with the composition information. The modeler can inspect this file and make changes as needed.

- The (possibly updated) composition information is used for inferring the names of all atomic species from the model. A template XML file is produced for the atomic refinement relation. The template contains a trivial refinement, namely the renaming of all atomic species by prepending “r” to their original names. The modeler must edit this file in order to describe a nontrivial refinement.

- The composition information and the atomic refinement relation are used for generating the refinement of complex species, using the approach presented in Section 3.3. The result is presented in XML format and contains the name and composition information for each of the refined species. The modeler can update this file to rename species, or to remove some of them, so as to match prior knowledge about the system that is modeled (some of the automatically generated combinations of refined atomic species may be impossible).

- The refinement of complex species is used to generate the refined reactions of the model, as described in Section 3.1. Again, the modeler can alter the results to remove some of the reactions.
- The refined model is generated as an SBML file, including all the intermediary information that was generated by the software.

The software was implemented in Python and uses Qt4 for the graphical user interface. It can be found at [36]

6 Quantitative refinement in other formalisms

Quantitative refinement is by no mean restricted to the reaction-based and ODE-based models. In this section we discuss the refinement in three other formalisms, namely rule-based models, Petri net models and guarded command based models. In each part we briefly introduce the modeling in that specific framework and we also give a short explanation on how to apply the refinement in each formalisms. The structural part of the refinement has a different solution in each approach, in some cases leading to a compact representation of the refined models. For a more detailed discussion we refer to [21].

6.1 Rule-based models refinement

A model within a a rule-based modelling framework is described by the molecules of interest, their components (i.e. a post-translational modification site) and the states corresponding to each component. The interaction between the components are captured through graph-rewriting rules, where a rule can refer to either a certain type of reactions or a class of reactions. We refer to [8] for a detailed presentation of this framework.

Rule-based languages are used to characterize the dynamics of the system at hand. Rules produce reactants introducing classes of reactions, which express classes of reactions describing specific interactions between atomic and/or complex species. In practice, a rule specifies *group rules*, which characterize interactions between species through regular expressions. The conversion from reactants to products is enabled through a rate law.

A graphical representation of the dimerization of EGF-EGFR is in Fig. 1.

In case one would need to refine either of the species to include two types of ligands or four types of receptors as discussed in Section 4.2, the only required

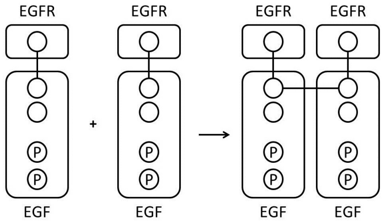

Figure 1: A graphical representation of the species EGF-EGFR and of the rule showing the dimerization of EGF-EGFR through a binding site . Note that the sites coloured in green represent binding sites, while the sites denoted by letter P represent phosphorylation sites.

adjustment needed is adding a site for EGF (with two possible vlaues) and one for EGFR (with four possible values). Note that the rule illustrated in Fig. 1 remains unchanged, in stark contrast with the combinatorial explosion discussed in Section 4.2.

6.2 Refinement of Petri net models

The *Petri nets* formalism is used to represent systems with concurrency and resource sharing, which makes it suitable for modeling biological systems. In this formalism each species is represented by a *place* with as many *tokens* as the number of instances of the species present in the system, and each reaction by a *transition* whose *pre-* and *post-places* correspond to the species on the left and the right hand side of the reaction, respectively, where arc multiplicities represent the stoichiometric coefficients of species involved in the reaction. For more information on modeling biological systems in the framework of Petri nets we refer to [26]. A Petri net model of the heat shock response case study is presented in Fig. 2.

Coloured Petri nets are an extension of Petri nets where places are assigned data types called color sets, and each place may host tokens of different colors (values in the place's color set). Transitions can have additional constraints on the colors of the tokens traversing them, in the form of *guards*, and arc multiplicities are replaced by expressions containing variables and/or values from the color set of the place connected to the arc. For more information on modeling with colored Petri nets, see [23].

Refinement of a model in the sense of Section 3 can be implemented in the framework of Petri nets by creating a new model where each refined species is represented as a place, and each refined reaction as a transition, which results in a model explosion of the same magnitude as in the case of reaction-based models. In the framework of colored Petri nets, the initial model can be transformed in the refined model via coloring. All subspecies of a species may be modeled using the same place as the parent species, having a color set with as many colors as the

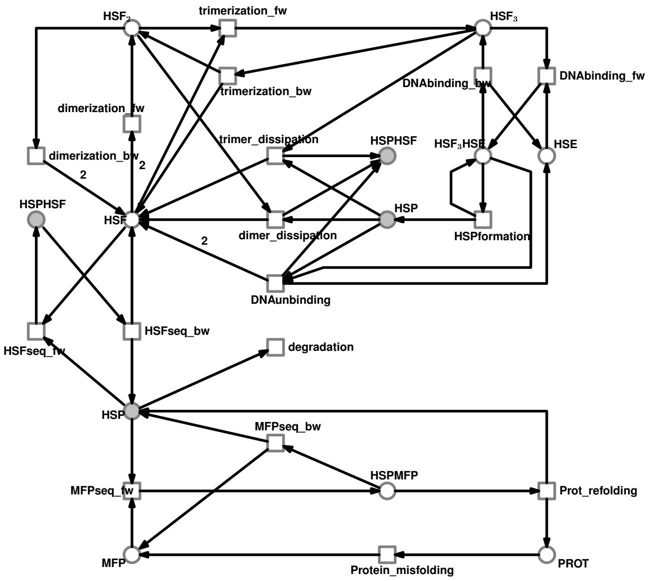

Figure 2: Petri net representation of the initial HSR model.

desired number of subspecies. New reactions can be represented with the same transition as the parent reaction, with possible constraints expressed via transition guards. It can also happen that the coloring scheme chosen for the refinement of species prompts to adding new transitions in the refined model to account for some of the refined reaction, in case the modeler wants to avoid too complex transition guards.

For the case study of the heat shock response, one coloring strategy is to consider integer color sets with as many colors as the number of subspecies for the species refining to at least two subspecies. A possible resulting model is depicted in Fig. 3. Compared to the network for the initial HSR model, this refined version contains several additional transitions. They account for reactions that have the same left hand side, but different right hand sides, e.g. a trimer with one acetylated molecule can produce either a non-acetylated monomer and a single-acetylated dimer, or an acetylated monomer and a non-acetylated dimer. The same model could be implemented while preserving the network structure by using variables on each adjacent arc and a guard on the trimerization$_b$w transition that accounts for all valid variable bindings at once.

A different coloring strategy is to consider the color set of places representing complex species to be the cartesian product of the color sets of the places representing the components of the complex. In this case, there is a distinction between e.g. one-acetylated dimers, depending on which of the two composing monomers is acetylated. This results in an adjustment of the kinetic constants of some reactions, but the resulting colored Petri net has exactly the same structure as the initial one. We refer to [14, 21] for all details of this construction. Note that both colored Petri net representations of the refinement are smaller in size than

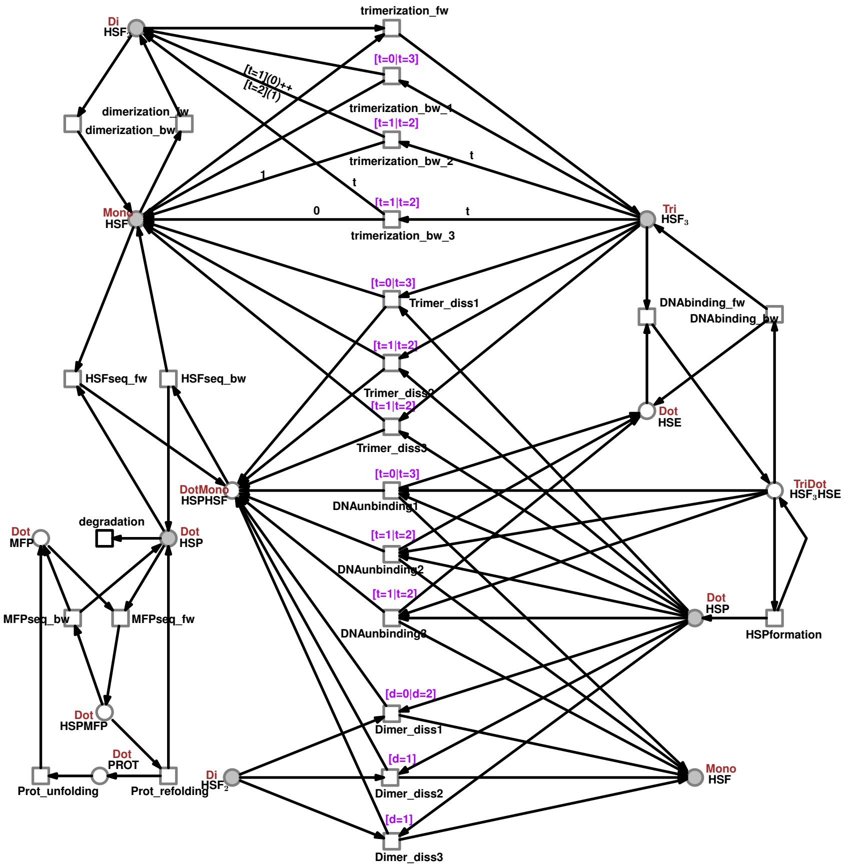

Figure 3: Representation of the refined HSR model as a colored Petri net, using as few colors as possible.

the fully expanded model, showing that the framework of colored Petri nets can be successfully used to obtain compact models upon refinement.

6.3 Guarded command-based models refinement

The *guarded command-based models*, inspired by the guarded command languages first introduced in [7], is a modelling framework to capture the dynamics of alternative and repetitive constructs with a non-deterministic component in which the enabled activity is not utterly dependent on the initial input.

A guarded command-based model comprises a set of variables and a set of guarded commands. A guarded command consists of a *guard*, an *update* and a corresponding *rate* to the guarded command. The guard is a Boolean predicate over all the variables in the model and the update describes a transition which the system can make if the guard is true. To obtain the guarded command corresponding to a reaction of a reaction network we use the approach proposed in [1], for example the guard corresponding to the reaction $\mathsf{F} + \mathsf{F} \to \mathsf{F}$ of Table 2 is obtained

Table 3: Basic guarded command-based model for intermediate filaments self-assembly.

Guarded command	
$[r_2]\, \mathsf{T} \geq 2 \rightarrow \mathsf{T}^2 * \mathsf{k_t} : \mathsf{T}' = \mathsf{T} - 2 \wedge \mathsf{O}' = \mathsf{O} + 1;$	(2)
$[r_3]\, \mathsf{O} \geq 2 \rightarrow \mathsf{O}^2 * \mathsf{k_o} : \mathsf{O}' = \mathsf{O} - 2 \wedge \mathsf{H}' = \mathsf{H} + 1;$	(3)
$[r_4]\, \mathsf{H} \geq 2 \rightarrow \mathsf{H}^2 * \mathsf{k_h} : \mathsf{H}' = \mathsf{H} - 2 \wedge \mathsf{U}' = \mathsf{U} + 1;$	(4)
$[r_5]\, \mathsf{U} \geq 2 \rightarrow \mathsf{U}^2 * \mathsf{k_h} : \mathsf{U}' = \mathsf{U} - 2 \wedge \mathsf{F}' = \mathsf{F} + 1;$	(5)
$[r_6]\, \mathsf{U} \geq 1 \wedge \mathsf{F} \geq 1 \rightarrow \mathsf{U} * \mathsf{F} * \mathsf{k_{uf}} : \mathsf{U}' = \mathsf{U} - 1 \wedge \mathsf{F}' = \mathsf{F} - 1 \wedge \mathsf{F}' = \mathsf{F} + 1$	(6)
$[r_7]\, \mathsf{F} \geq 2 \rightarrow \mathsf{F}^2 * \mathsf{k_{ff}} : \mathsf{F}' = \mathsf{F} - 2 \wedge \mathsf{F}' = \mathsf{F} + 1.$	(7)

as follows:

- the reaction can be enabled whenever there are at least two F in the system to bind and form an F, therefore, we define the corresponding *guard* to be "$\mathsf{F} \geq 2$", i.e. the guarded command can be enabled whenever $\mathsf{F} \geq 2$;
- we define the *rate* corresponding to the guarded command to be "$\mathsf{F}^2 * \mathsf{k_{ff}}$" which is in correspondence with the definition of a reaction rate of a mass-action ODE-based model, see [25];
- we define the *update* corresponding to the guarded command to be "$\mathsf{F}' = \mathsf{F} - 2 \wedge \mathsf{F}' = \mathsf{F} + 1$", i.e. whenever the guard is enable two F are consumed and one F is produced.

The list of all guarded commands corresponding to the basic intermediate filaments self-assembly is presented in Table 3.

Refinement in guarded command-based models is similar to the one of reaction-based models. In this approach whenever there is a refined variable in a guard we replace that guard with a set of guards considering to all possible refinements whereas in the refinement of reaction networks we would replace each reaction involving any refined reactant by the corresponding set of all possible refined reactions, for more information we refer to [1].

The list of all guarded commands corresponding to the refined intermediate filaments self-assembly is presented in Table 4.

7 Discussion

We discussed in this paper quantitative model refinement, an algorithmic approach for building large biomodels in an iterative fashion, while ensuring that the fit and the validation of the model is preserved throughout the construction. This allows the computational modeler to avoid repeating parameter estimation in each step of the model construction, even as the model size increases in each step; rather, the modeler may choose a setup that allows the model to preserve its fit to existing data in each step. Quantitative refinement also allows the modeler to deal with partial information about some of the parameters of the model, including such numerical values of the parameters whenever available, checking their consistency with the other parameters and with the data, and compensating for lack of information about parameters with an algorithmic solution. We investigated the versatility of the fit-preserving refinement method with respect to four broadly used frameworks: reaction models, rule-based models , Petri net models, and guarded command-based models.

The computational advantages of the refinement-driven top-down approach as opposed to the bottom-up approach based on collection of submodels is most evident in the case study on the ErbB signaling pathway. For instance, the ErbbB model of [3], consisting of 828 reactions and 499 reactants, was fit to experimental data by running about 100 times annealing methods, over 24 hours on a cluster consisting of 100 nodes. The refinement-driven approach starts from an initial model of [17] consisting of 103 reactants and 148 reactions, and leading to a refined model consisting of 421 reactants involved in 928 reactions; the refined model has a good numerical behavior, avoiding any supplementary model fit.

An interesting challenge that remains open to investigate is the scalability of the quantitative model refinement approach on larger case studies.

Acknowledgments.

This work was partially supported by the Academy of Finland under project 267915. Bogdan Iancu's current affiliation is at Department of Mathematics and Statistics, University of Turku, Finland.

References

[1] Sepinoud Azimi and Ion Petre. Quantitative model refinement for guarded command models. *To appear*, 2015.

[2] James B Bassingthwaighte. Strategies for the physiome project. *Annals of biomedical engineering*, 28(8):1043–1058, 2000.

[3] William W. Chen, Birgit Schoeberl, Paul J. Jasper, Mario Niepel, Ulrik B. Nielsen, Douglas A. Lauffenburger, and Peter K. Sorger. Input–output behavior of erbb signaling pathways as revealed by a mass action model trained against dynamic data. *Molecular Systems Biology*, 5:239, 2009.

[4] Gheorghe Craciun and Casian Pantea. Identifiability of chemical reaction networks. *Journal of Mathematical Chemistry*, 44(1):244–259, 2008.

[5] Elena Czeizler, Eugen Czeizler, Bogdan Iancu, and Ion Petre. Quantitative model refinement as a solution to the combinatorial size explosion of biomodels. *Electronic Notes in Theoretical Computer Science*, 284:35–53, 2012.

[6] Eugen Czeizler, Andrzej Mizera, Elena Czeizler, Ralph-Johan Back, John E Eriksson, and Ion Petre. Quantitative analysis of the self-assembly strategies of intermediate filaments from tetrameric vimentin. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 9(3):885–898, 2012.

[7] Edsger W Dijkstra. Guarded commands, nondeterminacy and formal derivation of programs. *Communications of the ACM*, 18(8):453–457, 1975.

[8] James R Faeder, Michael L Blinov, Byron Goldstein, and William S Hlavacek. Rule-based modeling of biochemical networks. *Complexity*, 10(4):22–41, 2005.

[9] Cristian Gratie and Ion Petre. Fit-preserving data refinement of mass-action reaction networks. In Arnold Beckmann, Erzsebet Csuhaj-Varju, and Klaus Meer, editors, *Language, Life, Limits*, volume 7938 of *Lecture Notes in Computer Science*, pages 204–213. Springer, 2014.

[10] Cristian Gratie and Ion Petre. Fit-preserving data refinement of mass-action reaction networks. Technical report, Turku Centre for Computer Science, 2014.

[11] Cristian Gratie and Ion Petre. Complete characterization for the fit-preserving data refinement of mass-action reaction networks. Technical Report 1128, Turku Centre for Computer Science, 2015.

[12] D.-E. Gratie, B. Iancu, S. Azimi, and I. Petre. Quantitative model refinement in four different frameworks, with applications to the heat shock response. Technical Report 1067, TUCS, 2013.

[13] Diana-Elena Gratie, Bogdan Iancu, and Ion Petre. Ode analysis of biological systems. In Marco Bernardo, Erik de Vink, Alessandra Di Pierro, and Herbert Wiklicky, editors, *Formal Methods for Dynamical Systems*, number 7938 in Lecture Notes in Computer Science, pages 29–62. Springer, 2013.

[14] Diana-Elena Gratie and Ion Petre. Hiding the combinatorial state space explosion of biomodels through colored petri nets. *Annals of University of Bucharest*, LXI:23–41, 2014.

[15] Harald Herrmann and Ueli Aebi. Intermediate filaments: molecular structure, assembly mechanism, and integration into functionally distinct intracellular scaffolds. *Annual review of biochemistry*, 73(1):749–789, 2004.

[16] Stefan Hoops, Sven Sahle, Ralph Gauges, Christine Lee, Jürgen Pahle, Natalia Simus, Mudita Singhal, Liang Xu, Pedro Mendes, and Ursula Kummer. Copasi—a complex pathway simulator. *Bioinformatics*, 22(24):3067–3074, 2006.

[17] J.J. Hornberg, B. Binder, F.J. Bruggeman, B. Schoeberl, R. Heinrich, and H.V. Westerhoff. Control of MAPK signalling: from complexity to what really matters. *Oncogene*, 24(36):5533–5542, 2005.

[18] M. Hucka and et al. The systems biology markup language (sbml): a medium for representation and exchange of biochemical network models. *Bioinformatics*, 19(4):524–531, 2003.

[19] B. Iancu, El. Czeizler, Eu. Czeizler, and I. Petre. Quantitative refinement of reaction models. *International Journal of Unconventional Computing*, 8(5-6):529–550, 2012.

[20] Bogdan Iancu. *Quantitative Refinement of Reaction-Based Biomodels*. PhD thesis, Turku Centre for Computer Science, 2015.

[21] Bogdan Iancu, Diana-Elena Gratie, Sepinoud Azimi, and Ion Petre. On the implementation of quantitative model refinement. In *Algorithms for Computational Biology*, pages 95–106. Springer, 2014.

[22] Nebiat Ibssa. Quantitative model refinement, 2015.

[23] Kurt Jensen and Lars M. Kristensen. *Coloured Petri Nets*. Springer-Verlag Berlin Heidelberg, 2009.

[24] Jonathan R Karr, Jayodita C Sanghvi, Derek N Macklin, Miriam V Gutschow, Jared M Jacobs, Benjamin Bolival, Nacyra Assad-Garcia, John I Glass, and Markus W Covert. A whole-cell computational model predicts phenotype from genotype. *Cell*, 150(2):389–401, 2012.

[25] Edda Klipp, Ralf Herwig, Axel Kowald, Christoph Wierling, and Hans Lehrach. *Systems biology in practice: concepts, implementation and application*. John Wiley & Sons, 2008.

[26] Ina Koch, Wolfgang Reisig, and Falk Schreiber. *Modeling in systems biology: the Petri net approach*, volume 16. Springer Science & Business Media, 2010.

[27] Elias Lazarides. Intermediate filaments as mechanical integrators of cellular space. *Nature*, 283(5744):249–256, 1980.

[28] Andrzej Mizera, Eugen Czeizler, and Ion Petre. Self-assembly models of variable resolution. In *Transactions on Computational Systems Biology XIV*, pages 181–203. Springer, 2012.

[29] Elaine Murphy, Vincent Danos, Jérôme Feret, Jean Krivine, and Russell Harmer. *Elements of Computational Systems Biology*, chapter Rule Based Modelling and Model Refinement, pages 83–114. Wiley Book Series on Bioinformatics. John Wiley & Sons, Inc., 2010.

[30] Grégoire Nicolis and Ilya Prigogine. *Self-Organizationin Nonequilibrium Systems: From Dissipative Structures to Order Through Fluctuations*. Wiley, 1977.

[31] Kristian Ovaska and et al. Large-scale data integration framework provides a comprehensive view on glioblastoma multiforme. *Genome medicine*, 2(9):65+, September 2010.

[32] I. Petre, A. Mizera, C.L. Hyder, A. Meinander, A. Mikhailov, R.I. Morimoto, L. Sistonen, J.E. Eriksson, and R.J. Back. A simple mass-action model for the eukaryotic heat shock response and its mathematical validation. *Natural Computing*, 10(1):595–612, 2011.

[33] Marco Pistore and Davide Sangiorgi. A partition refinement algorithm for the π-calculus. In *Computer Aided Verification*, pages 38–49. Springer, 1996.

[34] Yoram Rudy. From genome to physiome: integrative models of cardiac excitation. *Annals of biomedical engineering*, 28(8):945–950, 2000.

[35] M. Schliwa. *The Cytoskeleton*, volume 13 of *Cell Biology Monographs*. Springer-Verlag, Vienna, Austria, 1986.

[36] Fatemeh Shokri. Structref, 2015.

[37] Ichiro Suzuki and Tadao Murata. A method for stepwise refinement and abstraction of petri nets. *Journal of computer and system sciences*, 27(1):51–76, 1983.

Table 4: Refined guarded command-based model for intermediate filaments self-assembly.

Guarded command	
$[r_8]\ \mathsf{rT} \geq 2 \rightarrow \mathsf{rT}^2 * \mathsf{k_t}' : \mathsf{rT}' = \mathsf{rT} - 2 \wedge \mathsf{rO}' = \mathsf{rO} + 1;$	(8)
$[r_9]\ \mathsf{rO} \geq 2 \rightarrow \mathsf{rO}^2 * \mathsf{k_o}' : \mathsf{rO}' = \mathsf{rO} - 2 \wedge \mathsf{rH}' = \mathsf{rH} + 1;$	(9)
$[r_{10}]\ \mathsf{rH} \geq 2 \rightarrow \mathsf{rH}^2 * \mathsf{k_h}' : \mathsf{rH}' = \mathsf{rH} - 2 \wedge \mathsf{rF_1}' = \mathsf{rF_1} + 1;$	(10)
$[r_{11}]\ \mathsf{rF_1} \geq 2 \rightarrow \mathsf{rF_1}^2 * \mathsf{k_{(1,1)}}' : \mathsf{rF_1}' = \mathsf{rF_1} - 2 \wedge \mathsf{rF_2}' = \mathsf{rF_2} + 1;$	(11)
for $1 \leq \mathsf{i} \leq 4$:	
$[r_{12}]\ \mathsf{rF_1} \geq 1 \wedge \mathsf{rF_i} \geq 1 \wedge \rightarrow \mathsf{rF_1} * \mathsf{rF_i} * \mathsf{k_{(1,i)}}' : \mathsf{rF_1}' = \mathsf{rF_1} - 1 \wedge$ $\mathsf{rF_i}' = \mathsf{rF_i} - 1 \wedge \mathsf{rF_{i+1}}' = \mathsf{rF_{i+1}} + 1;$	(12)
for $\mathsf{j} \in \{5, \geq 6\}$:	
$[r_{13}]\ \mathsf{rF_1} \geq 1 \wedge \mathsf{rF_j} \geq 1 \wedge \rightarrow \mathsf{rF_1} * \mathsf{rF_j} * \mathsf{k_{(1,i)}}' : \mathsf{rF_1}' = \mathsf{rF_1} - 1 \wedge$ $\mathsf{rF_j}' = \mathsf{rF_j} - 1 \wedge \mathsf{rF_{\geq 6}}' = \mathsf{rF_{\geq 6}} + 1;$	(13)
$[r_{14}]\ \mathsf{rF_2} \geq 2 \rightarrow \mathsf{rF_2}^2 * \mathsf{k_{(2,2)}}' : \mathsf{rF_2}' = \mathsf{rF_2} - 2 \wedge \mathsf{rF_4}' = \mathsf{rF_4} + 1;$	(14)
for $\mathsf{i} \in \{3, 4, 5, \geq 6\}$:	
$[r_{15}]\ \mathsf{rF_i} \geq 2 \rightarrow \mathsf{rF_1} * \mathsf{rF_j} * \mathsf{k_{(1,i)}}' : \mathsf{rF_i}' = \mathsf{rF_i} - 2 \wedge \mathsf{rF_{\geq 6}}' = \mathsf{rF_{\geq 6}} + 1;$	(15)
$[r_{16}]\ \mathsf{rF_2} \geq 1 \wedge \mathsf{rF_3} \geq 1 \wedge \rightarrow \mathsf{rF_2} * \mathsf{rF_3} * \mathsf{k_{(2,3)}}' : \mathsf{rF_2}' = \mathsf{rF_2} - 1 \wedge$ $\mathsf{rF_3}' = \mathsf{rF_3} - 1 \wedge \mathsf{rF_5}' = \mathsf{rF_5} + 1;$	(16)
for $2 \leq \mathsf{i} < \mathsf{j} \leq 5, \mathsf{i} + \mathsf{j} \geq 6$:	
$[r_{17}]\ \mathsf{rF_i} \geq 1 \wedge \mathsf{rF_j} \geq 1 \wedge \rightarrow \mathsf{rF_i} * \mathsf{rF_j} * \mathsf{k_{(i,j)}}' : \mathsf{rF_i}' = \mathsf{rF_i} - 1 \wedge$ $\mathsf{rF_j}' = \mathsf{rF_j} - 1 \wedge \mathsf{rF_{\geq 6}}' = \mathsf{rF_{\geq 6}} + 1;$	(17)
$[r_{18}]\ \mathsf{rF_i} \geq 1 \wedge \mathsf{rF_{\geq 6}} \geq 1 \wedge \rightarrow \mathsf{rF_i} * \mathsf{rF_{\geq 6}} * \mathsf{k_{(i,\geq 6)}}' : \mathsf{rF_i}' = \mathsf{rF_i} - 1 \wedge$ $\mathsf{rF_{\geq 6}}' = \mathsf{rF_{\geq 6}} - 1 \wedge \mathsf{rF_{\geq 6}}' = \mathsf{rF_{\geq 6}} + 1.$	(18)

TURKU CENTRE *for* COMPUTER SCIENCE

Joukahaisenkatu 3-5 A, 20520 TURKU, Finland | www.tucs.fi

University of Turku
Faculty of Mathematics and Natural Sciences
- Department of Information Technology
- Department of Mathematics and Statistics

Turku School of Economics
- Institute of Information Systems Sciences

Åbo Akademi University
- Computer Science
- Computer Engineering

ISBN 978-952-12-3325-8
ISSN 1239-1891