Diana-Elena Gratie | Ion Petre

Full structural model refinement as type refinement of colored Petri nets

Turku Centre *for* Computer Science

TUCS Technical Report
No 1135, April 2015

Full structural model refinement as type refinement of colored Petri nets

Diana-Elena Gratie
Turku Centre for Computer Science
Åbo Akademi University, Department of Computer Science
Joukahaisenkatu 3-5A, 20520 Turku, Finland
dgratie@abo.fi

Ion Petre
Turku Centre for Computer Science
Åbo Akademi University, Department of Computer Science
Joukahaisenkatu 3-5 A, 20520 Turku, Finland
ipetre@abo.fi

TUCS Technical Report
No 1135, April 2015

Abstract

In this paper we propose a method for implementing a full structural model refinement of a (biological) model represented as a (colored) Petri net. We build on the full structural data refinement definition of C. Gratie and Petre, and the type refinement of colored Petri nets introduced by Charles Lakos. Given a (biological) reaction-based model and a desired full structural refinement of it, we propose a general coloring scheme for a colored Petri net implementation of the model and give an algorithm for adding the refinement details in the Petri net model. We then prove that the construction is a type refinement, and that by our choice of color sets the resulting refined colored Petri net implements the full structural refinement of the given model.

Keywords: Colored Petri nets, type refinement, reaction network, structural model refinement.

TUCS Laboratory
Computational Biomodelling Laboratory

1 Introduction

Model refinement, the process of adding more details to an existing model, is an important step in the model building cycle. Many refinement methods have been proposed for different modeling frameworks and formalisms, e.g., action systems [1], Petri nets [16, 10], kappa [4], biochemical reaction networs [6], π-calculus [15], etc. We bridge here two modelling frameworks and their respective ways of implementing refinement, namely reaction network models with structural refinement and colored Petri nets with type refinement.

Type refinement of colored Petri nets has been introduced in [10], and consists of refining the color sets of places such that the new color sets are polymorphic with the initial color sets. The authors see this as adding some supplementary data to a given data type represented as a color set, e.g. include in the entry of a book in a library not only its title and authors, but also the maximum number of days it can be borrowed.

The concept of *(full) structural refinement* of a reaction network (bio-)model has been introduced in [6] (where it was called data refinement), with a focus on an ODE-based representation of a model and its refinement. A sufficient condition for the refined model to preserve the fit of the original one was discussed in [5] for mass-action models. We follow in this paper the terminology of [5]. We use the main concepts of *species refinement* and *(full) structural refinement* for models represented as (colored) Petri nets, and give a methodology for implementing full structural refinements as type refinements of colored Petri nets. An approach to implementing model refinement in the colored Petri net framework has been exemplified for a model of the eukaryotic heat shock response mechanism in [7]. The authors present there two coloring schemes that can be used for the particular refinement they were implementing. We derive here a general coloring scheme for model refinement that can be used when implementing a full structural data refinement of a model.

We assume the reader is familiar with (colored) Petri nets, but we recall some of the basic definitions so that the paper is self-contained.

The paper is structured as follows: in Section 2 we present reaction network (also called reaction-based) models and the notions of *species refinement* and *(full) structural refinement* of such models, with a discussion on the explosion of the model induced by a refinement, in terms of number of species and reactions that the initial model refines to. In Section 3 we recall some notions and notations for Petri nets and their colored version, give a coloring scheme and discuss how a reaction network model can be implemented as a (colored) Petri net. We continue in Section 4 with proposing a type refinement based on a refinement relation ρ and prove that the chosen type refinement results in a colored Petri net that is the implementation of the full structural $\rho-$refinement of the initial model. We draw our conclusions and discuss about the model size and successive refinements in Section 5.

2 Model refinement

In systems biology, model refinement comprises two aspects: the structural side and the quantitative side. The structural side handles the newly introduced species and presents a methodology for computing the new set of reactions, while the quantitative side deals with changes in the kinetic constants of the model and ways of setting the new parameters in such a way that previous data is used. *Quantitative model refinement* was introduced in [14, 4] for rule-based models, and for reaction-based models in [12, 6]. We recall here the structural refinement of reaction network models, as presented in [6] and based on the terminology of [5]. We are only interested in the structural refinement, so we will not focus on any quantitative details.

A *reaction-based model* M consists of a finite set of *species* $\mathscr{S} = \{A_1, \ldots, A_m\}$ and a finite set of reactions *reactions* $\mathscr{R} = \{r_1, \ldots, r_n\}$ using only species in $\mathscr{S}$. A reaction $r_j \in \mathscr{R}$ can be formulated as a rewriting rule of the form:

$$r_j \ : \ c_{1,j}A_1 + \ldots + c_{m,j}A_m \xrightarrow{k_{r_j}} c'_{1,j}A_1 + \ldots + c'_{m,j}A_m, \tag{1}$$

with the meaning that $c_{i,j}$ copies of species A_i are consumed by the reaction and $c'_{i,j}$ copies of species A_i are produced, $i = 1..m$. Constants $c_{1,j}, \ldots, c_{m,j}, c'_{1,j}, \ldots, c'_{m,j} \in \mathbb{N}$ are the *stoichiometric coefficients* of r_j and $k_{r_j} \geq 0$ is the *kinetic rate constant* of reaction r_j. We denote by $\mathbf{r_j^-} = (c_{1,j}, \ldots, c_{m,j})$ the vector of stoichiometric coefficients on the left hand side of reaction, for the species being consumed in reaction r_j, and by $\mathbf{r_j^+} = (c'_{1,j}, \ldots, c'_{m,j})$ the vector of stoichiometric coefficients on its right hand side, those of species being produced. Without a risk of ambiguity, reaction r_j can then be written as $\mathbf{r_j^-} \xrightarrow{k_{r_j}} \mathbf{r_j^+}$.

Example 1. *A biological system with two reversible reactions that encode the dimerization of a molecule P can be represented as a reaction-based model $M = (\mathscr{S}, \mathscr{R})$ where $\mathscr{S} = \{P, P_2\}$ and $\mathscr{R} = \{2P \to P_2,\ P_2 \to 2P\}$. P represents the monomeric molecule and P_2 is the dimer that is formed from two P monomers.*

Data refinement is the type of refinement of a model that consists in adding details related to the species of the model, i.e., it replaces a species with several of its subspecies. The subspecies may account for post-translational modifications of macromolecules, or distinguish between possible variants of some trait.

All species are considered to be refined at once, thus each species in an initial model is replaced by a non-empty set of refined species to yield a refined model, as dictated by a *species refinement relation* ρ. This is formalized in Definition 1.

Definition 1 ([5])**.** *Given two sets of species $\mathscr{S}$ and $\mathscr{S}'$, and a relation $\rho \subseteq \mathscr{S} \times \mathscr{S}'$, we say that ρ is a* species refinement relation *iff it satisfies the following conditions:*

1. *for each $A \in \mathscr{S}$ there exists $A' \in \mathscr{S}'$ such that $(A, A') \in \rho$;*

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

We denote $\rho(A) = \{A' \in \mathscr{S}' \mid (A, A') \in \rho\}$. *We say that all species* $A' \in \rho(A)$ *are* siblings.

Intuitively, each species $A \in \mathscr{S}$ is replaced in the refined model with the set of species $\rho(A)$. For the case where $\rho(A)$ is a singleton set, one may consider that species A does not change, even if its refined counterpart is denoted by a different name in $\mathscr{S}'$; such a refinement of a species is called *trivial*.

Next we recall the definitions of refinement of a vector (of stoichiometric coefficients), of a reaction, and of a reaction-based model.

Definition 2 ([5]). *Let* $\mathscr{S} = \{A_1, \ldots, A_m\}$ *and* $\mathscr{S}' = \{A'_1, \ldots, A'_p\}$ *be two sets of species, and* $\rho \subseteq \mathscr{S} \times \mathscr{S}'$ *a species refinement relation.*

1. *Let* $\alpha = (\alpha_1, \ldots, \alpha_m) \in \mathbb{N}^{\mathscr{S}}$ *and* $\alpha' = (\alpha'_1, \ldots, \alpha'_p) \in \mathbb{N}^{\mathscr{S}'}$. *We say that* α' *is a* ρ-refinement *of* α *if*

$$\sum_{\substack{1 \le j \le p \\ A'_j \in \rho(A_i)}} \alpha'_j = \alpha_i, \text{ for all } 1 \le i \le m.$$

   We denote by $\rho(\alpha)$ *the set of all* $\rho-$*refinements of* α.

2. *Let* $r : r^- \to r^+$ *and* $r' : r'^- \to r'^+$ *be two reactions over* $\mathscr{S}$ *and* $\mathscr{S}'$, *resp. We say that* r' *is a* ρ-refinement *of* r *if*

$$r'^- \in \rho(r^-) \text{ and } r'^+ \in \rho(r^+).$$

   We denote by $\rho(r)$ *the set of all* $\rho-$*refinements of* r. *Note that* $\rho(r) = \rho(r^-) \times \rho(r^+)$.

3. *Let* $M = (\mathscr{S}, \mathscr{R})$ *and* $M' = (\mathscr{S}', \mathscr{R}')$ *be two reaction-based models, and* $\rho \subseteq \mathscr{S} \times \mathscr{S}'$ *a species refinement relation. We say that* M' *is a* ρ-structural refinement of M *if*

$$\mathscr{R}' \subseteq \bigcup_{r \in \mathscr{R}} \rho(r) \text{ and } \rho(r) \cap \mathscr{R}' \neq \varnothing \; \forall r \in \mathscr{R}.$$

   In case $\mathscr{R}' = \bigcup_{r \in \mathscr{R}} \rho(r)$, *we say* M' *is the* full structural ρ-refinement *of* M, *denoted* $M' = M_\rho$.

Model explosion. Note that a vector of coefficients $\alpha' \in \mathbb{N}^{\mathscr{S}}$ that respects the sum condition $\sum_{\substack{1\leq j\leq p \\ A'_j\in\rho(A_i)}} \alpha'_j = \alpha_i$, for all $1 \leq i \leq m$ can be seen as a way of choosing α_i elements from a bag containing elements of $|\rho(A_i)|$ types, where the selection may contain several elements of the same type.

The total number of different ways in which one may choose k elements from a bag with elements of n types (assuming enough copies of each type are available) is $\left(\!\binom{n}{k}\!\right) = \binom{n+k-1}{k}$, the so-called *multiset coefficient*.

A reaction r_j of the form (1) can refine to $\prod_{1\leq i\leq n} \left(\!\binom{|\rho(A_i)|}{c_{i,j}}\!\right) \cdot \left(\!\binom{|\rho(A_i)|}{c'_{i,j}}\!\right)$ different reactions. The number stems from the number of possible ways of choosing $c_{i,j}$ ($c'_{i,j}$, resp.) copies from the possible refinements of a species $A_i \in \mathscr{S}$. The number of reactions in a full structural $\rho-$refinement of a model is thus:

$$\sum_{1\leq j\leq m} \prod_{1\leq i\leq n} \left(\!\binom{|\rho(A_i)|}{c_{i,j}}\!\right) \cdot \left(\!\binom{|\rho(A_i)|}{c'_{i,j}}\!\right).$$

Example 2. *Consider the reaction-based model $M = (\mathscr{S}, \mathscr{R})$ from Example 1. One possible refinement for this model is to consider that molecule P can be in two states: acetylated ($P^{(1)}$) and non-acetylated($P^{(0)}$). Then the dimer P_2 could have none ($P_2^{(0)}$), one ($P_2^{(1)}$) or both ($P_2^{(2)}$) of its composing monomers acetylated. Consider a set $\mathscr{S}' = \{P^{(0)}, P^{(1)}, P_2^{(0)}, P_2^{(1)}, P_2^{(2)}, P_3^{(0)}\}$. A relation $\rho \subseteq \mathscr{S} \times \mathscr{S}'$ that would capture such a refinement is $\rho = \{(P, P^{(0)}), (P, P^{(1)}), (P_2, P_2^{(0)}), (P_2, P_2^{(1)}), (P_2, P_2^{(2)})\}$. One can easily see that ρ is a refinement relation, based on Definition 1.*

A full structural ρ-refinement of M is the model $M' = \mathscr{S}', \mathscr{R}'$ where $\mathscr{R}' = \{2P^{(0)} \to P_2^{(0)}, 2P^{(0)} \to P_2^{(1)}, 2P^{(0)} \to P_2^{(2)}, P^{(0)} + P^{(1)} \to P_2^{(0)}, P^{(0)} + P^{(1)} \to P_2^{(1)}, P^{(0)} + P^{(1)} \to P_2^{(2)}, 2P^{(1)} \to P_2^{(0)}, 2P^{(1)} \to P_2^{(1)}, 2P^{(1)} \to P_2^{(2)}, P_2^{(0)} \to 2P^{(0)}, P_2^{(0)} \to P^{(0)} + P^{(1)}, P_2^{(0)} \to 2P^{(1)}, P_2^{(1)} \to 2P^{(0)}, P_2^{(1)} \to P^{(0)} + P^{(1)}, P_2^{(1)} \to 2P^{(1)}, P_2^{(2)} \to 2P^{(0)}, P_2^{(2)} \to P^{(0)} + P^{(1)}, P_2^{(2)} \to 2P^{(1)}.\}$

3 Modeling biological systems as (colored) Petri nets

Many biological models are implemented as Petri nets due to the graphical, intuitive formalism, and the many simulation strategies they offer. We start our discussion over refinement and implementations of models as Petri nets from the standard version of Petri nets. We then continue with colored Petri nets.

3.1 Preliminaries

There exist two ways of defining colored Petri nets, one proposed by Kurt Jensen in [8], and an equivalent one adapted from the first definition, by Charles Lakos in [10]. In this paper we consider the definition of colored Petri nets proposed

by Lakos because it does not explicitly include transition guards (that we are not using in our construction) and because of the definition of type refinement of colored Petri nets proposed in [10]. We assume the reader is familiar with the basic notions and notations related to Petri nets and we refer to [13] for details. We also assume that the reader is familiar with constructing a standard Petri net associated to a reaction-based model; we refer to [2] for details.

Definition 3 ([10])**.** *A* colored Petri net *is a tuple* $N = (P, T, A, C, E, \Sigma, \mathbb{M}, \mathbb{Y}, M_0)$ *where:*

- P *is the finite set of places*
- T *is the finite set of transitions, such that* $P \cap T = \varnothing$
- $A \subseteq P \times T \cup T \times P$ *is the finite set of arcs*
- Σ *is a universe of non-empty color sets with an associated partial order*
- $C : P \cup T \to \Sigma$ *is the color set function, assigning color sets to places and (modes) of transitions*
- $E : A \to \Phi\Sigma$ *is the arc expression function, where* $E(p,t), E(t,p) : C(t) \to \mu C(p)$
- $\mathbb{M} = \mu\{(p,c) \mid p \in P, c \in C(p)\}$ *is the set of markings*
- $\mathbb{Y} = \mu\{(t,c) \mid t \in T, c \in C(t)\}$ *is the set of steps*
- M_0 *the initial marking, with* $M_0 \in \mathbb{M}$

Arc expressions may contain variables, which are seen as symbols whose value is determined by the color (mode) of the transition the arc is connected with.

For any colored Petri net there exists a standard Petri net that is behaviorally equivalent, see [9]. The process of transforming a colored Petri net into its standard Petri net equivalent is called *unfolding*. We give in the following the definition of the unfolding of a colored Petri net as adapted from [9] to the notations we use.

Definition 4 ([9])**.** *Given a colored Petri net* $N = (P, T, A, \Sigma, C, E, \mathbb{M}, \mathbb{Y}, M_0)$*, its unfolded Petri net is denoted by* $N^* = (P^*, T^*, A^*, f^*, M_0^*)$*, where:*

- $P^* = I_P$*;*
- $T^* = I_T$*;*
- $A^* = \{((p,c),(t,c')) \in P^* \times T^* \mid E((p,t))(c')\langle c\rangle > 0\} \cup \{((t,c'),(p,c)) \in T^* \times P^* \mid E((t,p))(c')\langle c\rangle > 0\}$*;*

- $f^*((p,c),(t,c')) = E((p,t))(c')\langle c\rangle, \forall((p,c),(t,c')) \in A^*$ *and* $f^*((t,c'),(p,c)) = E((t,p))(c')\langle c\rangle,\ \forall((t,c'),(p,c)) \in A^*$*;*
- $M_0^*((p,c)) = M_0(p,c)$.

In order to implement a reaction-based model as a Petri net, one represents each species via a place, and each reaction via a transition having as pre-places the places representing the reactants of the reaction, and as post-places the places representing the products of the reaction, with each arc expression being the stoichiometry of the represented species in that reaction. For details on modelling biological systems using Petri nets we refer to [2].

Definition 5 (Implementation of a reaction network model as a Petri net)**.** *Given a reaction-based model $M = (\mathscr{S}, \mathscr{R})$, and a Petri net $N = (P, T, A, f, M_0)$ with $|\mathscr{S}| = |P|$ and $|\mathscr{R}| = |T|$, we say that the Petri net N* structurally implements *model M if there exists a bijection $\delta : \mathscr{S} \cup \mathscr{R} \to P \cup T$ mapping species of M into places of N and reactions of M into transitions of N ($\delta(x) \in P$, for all $x \in \mathscr{S}$ and $\delta(x) \in T$ for all $x \in \mathscr{R}$) such that for every reaction $r_j \in \mathscr{R}$ and its corresponding transition $t = \delta(r_j)$ and for every species $S_i \in \mathscr{S}$ the following conditions hold:*

1. *if $c_{i,j} > 0$ then $(\delta(S_i), t) \in A$ and $f(\delta(S_i), t) = c_{i,j}$, otherwise $(\delta(S_i), t) \notin A$*
2. *if $c'_{i,j} > 0$ then $(t, \delta(S_i)) \in A$ and $f(t, \delta(S_i)) = c'_{i,j}$, otherwise $(t, \delta(S_i)) \notin A$*

Example 3. *An example of a Petri net structural implementation of the model described in Example 1 is given in Figure 3. The bijection δ is defined such that $\delta(P) =$ P_, $\delta(P_2) =$ P_2, $\delta(2P \to P_2) =$ T_fw, $\delta(P_2 \to 2P) =$ T_bw. One can easily see that the arc multiplicities respect the two conditions in Definition 5.*

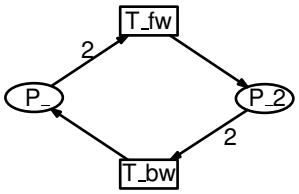


Figure 1: Standard Petri net implementation/representation of a dimerization model. Only multiplicities greater than 1 are displayed.

3.2 Coloring a standard Petri net

A colored Petri net representation of a model can be obtained from a standard Petri net implementation of the model by assigning to each place a color set with

just one element. We propose here a general coloring scheme that uses record color sets and can easily be extended to incorporate refinement details by adding new fields. Each place is assigned its own record color set with one field that has exactly one value. Each transition is assigned a color set that is a multiset of color sets of its pre- and post-places, where the multiplicity of each color set is given by the multiplicity of the arc connecting the place and the transition. It is basically a multiset with elements of different types. For example, the color set *CS_T_fw* in Figure 4 is a colection of two elements of type *CS_P* and one element of type *CS_P2*. Note that this is not the only possible coloring scheme and moreover it may not be optimal (in terms of number of variables and data structures used), but it is general. One may use integers, records, sets, Cartesian products, or whatever coloring scheme better suits the system being modeled.

A further change that is required when turning a standard Petri net into a colored one is assigning to each arc a with arc function $f(a) = k$ where $k \in \mathbb{N}$ the expression $E(a) = \mathsf{v}_1 + + \ldots + + \mathsf{v}_k$ where $++$ denotes multiset addition and $\mathsf{v}_i : C(p)$ are typed variables with $i = 1..k$, and p is the place of arc a. Intuitively, we use a different variable for each token that may traverse an arc. The total number of variables needed in a model is thus $\sum_{a \in A} f(a)$. A further change is in the initial marking, where each place p is assigned the same number of tokens as in the standard network, and all tokens have as color the one color in p's color set. We call such a colored Petri net the *trivial coloring* of the initial network.

We denote by $\mathcal{C}(x)$ the one color in the color set of a place/transtition x. In order to identify precisely the variables used in the expression of an arc $(x, y) \in A$ we denote the variables by $\mathsf{v}_{\mathsf{x,y,i}}$, where $i = 1..f((x, y))$. We also use the shorthand notation $\mathsf{v}_{\mathsf{a,i}}$ to denote the i-th variable on arc $a \in A$.

Definition 6 (Trivial coloring of a Petri net)**.** *Given a standard Petri net* $N = (P, T, A, f, M_0)$*, we call a* trivial coloring of N *a colored Petri net* $T(N) = (P, T, A, \Sigma, C, E, \mathbb{M}, \mathbb{Y}, M_0')$ *such that:*

- $\Sigma = \bigcup_{p \in P} C_p \cup \bigcup_{t \in T} C_t$ *where* $C_t : \{C_p \mid p \in P\} \to \mathbb{N}$ *is a multiset such that:*

$$C_t(C_p) = \begin{cases} 0 & (p,t) \notin A \text{ and } (t,p) \notin A \\ f((p,t)) & (p,t) \in A \text{ and } (t,p) \notin A \\ f((t,p)) & (p,t) \notin A \text{ and } (t,p) \in A \\ f((p,t)) + f((t,p)) & \text{otherwise} \end{cases}$$

- $C : P \cup T \to \Sigma$*, such that* $C(x)$ *is a record color set defined as above if* $x \in P$ *and a multiset defined as above if* $x \in T$

- $E(a) = \mathsf{v}_{\mathsf{a},1} + + \cdots + + \mathsf{v}_{\mathsf{a,f(a)}}$*, for all* $a \in A$*, where* $\mathsf{v}_{\mathsf{a,i}} : C(p)$ *with* p *being the place of arc* a

- $\mathbb{M}$ *is the set of markings*
- $\mathbb{Y}$ *is the set of steps*
- $M'_0(p) = M_0(p)\grave{}\mathcal{C}(p)$*, for all* $p \in P$

Example 4. *An example of a trivial coloring of the Petri net described in Example 3 is given in Figure 4.*

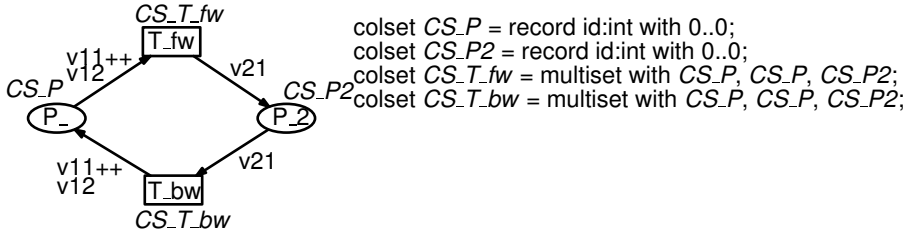


Figure 2: Trivial coloring of a Petri net implementation/representation of a dimerization model.

Definition 7 (Implementation of a reaction-based model as a colored Petri net)**.** *We say that a colored Petri net* N structurally implements *a given reaction-based model* M *iff the unfolding of* N *structurally implements model* M *in the sense of Definition 5.*

Proposition 1. *The unfolding of a trivial coloring of a standard Petri net* N *is equivalent to the initial net* N *(as every color set has exactly one color).*

Proposition 2. *If a standard Petri net* N *structurally implements a reaction-based model* M*, then its trivial coloring* $T(M)$ *structurally implements the same model* M*.*

Proof. By Proposition 1, N and $T(N)^*$ are equivalent, thus the unfolding of $T(N)$ structurally implements model M and, by Definition 7, $T(M)$ structurally implements M. □

3.3 Type refinement of colored Petri nets

Refinements of Petri nets have been a subject of interest for many years. In particular, we are concerned here with the work of Charles Lakos, who has identified and formalized three types of refinements: *type refinement*, *subnet refinement* and *node refinement*, see [10] for details. The concepts of type and node refinement have been further extended by Choppy et. al., see [3]. We prove in this paper that a full structural refinement of a model can be implemented via a type refinement of the colored Petri net representing the model.

We recall now the definition of type refinement of a colored Petri net as it was proposed in [10]. Σ denotes a universe of non-empty color sets with an associated partial order $<: \subseteq \Sigma \times \Sigma$ indicating that values from one color set X with $X <: Y$ can be used in contexts expecting values of Y. Π_Y is a projection function mapping values of X into values of Y. $\Phi\Sigma = \{X \to Y \mid X, Y \in \Sigma\}$ denotes the functions over Σ, and $\mu\mathbf{X} = \{X \to \mathbb{N}\}$ denotes the multisets over X. $E^-, E^+ : \mathbb{Y} \to \mathbb{M}$ represent the *incremental negative and positive, resp. changes* of the occurrence of a step Y, and are given by the linear extension of: $E^-((t,c)) = \sum_{p \in P}\{p\} \times E((p,t))(c)$ and $E^+((t,c)) = \sum_{p \in P}\{p\} \times E((t,p))(c), \forall t \in T, \forall c \in C(t)$.

Definition 8 ([10])**.** *Let N and N' be two colored Petri nets. A morphism $\Phi : N \to N'$ captures a* type refinement of a colored Petri net *if:*

1. *Φ is the identity function on P, T, A;*
2. *$C(x) <: \Phi(C)(x)$, for all $x \in P \cup T$;*
3. *$\Phi(1\,`(x,c)) = 1`(x, \Pi_{\Phi(C)(x)}(c))$ for all $x \in P \cup T$ and for all $c \in C(x)$;*
4. *$\Phi(E^-(1`(t,m)))(p) = \Pi_{\Phi(C)(p)}(E(p,t)(m)) = \Phi(E)(p,t)(\Pi_{\Phi(C)(t)}(m))$, for all $(p,t) \in A$ and for all $(t,m) \in \mathbb{Y}$;*
5. *$\Phi(E^+(1`(t,m)))(p) = \Pi_{\Phi(C)(p)}(E(t,p)(m)) = \Phi(E)(t,p)(\Pi_{\Phi(C)(t)}(m))$, for all $(t,p) \in A$ and for all $(t,m) \in \mathbb{Y}$.*

A morphism that captures a type refinement is a *system morphism*, see [10], which means that it is a *behavior-respecting* mapping of two colored Petri nets. Expressing structural refinement as a type morphism will thus guarantee that the behavior of the initial network is preserved in the refined network. Moreover, as discussed in [11], type refinement ensures bisimilarity between the initial and the refined network.

Note that for every refined state or action there exists a corresponding abstract state or action, resp. via the projection from subtype to supertype. Also note that in Definition 8, N denotes the refined network.

4 Full structural refinement as type refinement of colored Petri nets

In this section we prove that the full structural refinement of a reaction-based model implemented as a Petri net can be implemented as a type refinement of the trivial coloring of the Petri net. We give a coloring strategy (type refinement) for implementing a full structural data refinement of a model represented as a Petri net, and conclude by proving that our construction indeed implements the required full structural data refinement.

4.1 Implementing a full structural model refinement via a type refinement in a colored Petri net model

Intuitively, species refinement implies replacing each species with a non-empty set of species. This can be done in a colored Petri net by replacing for each place representing a species its default color set by a new record or enumeration color set having as many elements as the set of species that its corresponding species refines to. Or, for a definition using integer numbers, by a color set int 0..r where r is the cardinality of the refined subspecies set. Formally, we need to define a morphism from the refined colored Petri net to the initial colored Petri net that respects all the properties of a type refinement, as described in [10] and presented in Section 3.3.

Definition 9 (Colored Petri net implementation of a structural refinement of a reaction network model)**.** . *We say that a colored Petri net N structurally implements the full structural refinement of a model M as described by a refinement relation ρ iff the unfolding of N, N^*, structurally implements the full structural refinement of M, $\rho(M)$ in the sense of Definition 5.*

We describe next a type refinement of a given trivial coloring of a Petri net implementation of a reaction-based model M that captures the full structural data refinement of M as described by a given refinement relation ρ. The procedure builds the colored Petri net implementing the required full structural data refinement. Based on the colored Petri net built by this procedure we further detail the morphism Φ_ρ between the two nets.

Let $N = (P, T, A, \Sigma, C, E, \mathbb{M}, \mathbb{Y}, M_0)$ be a trivially colored Petri net that implements a reaction-based model $M = (\mathscr{S}, \mathscr{R})$ with correspondence function δ. Let $\rho \subseteq \mathscr{S} \times \mathscr{S}'$ be a full structural refinement relation that refines model M to model $M' = (\mathscr{S}', \mathscr{R}')$. We build a colored Petri net $N' = (P, T, A, \Sigma', C', E', \mathbb{M}', \mathbb{Y}', M_0')$ and then show that the construction is a type refinement. Moreover, we show that the resulting network implements the full structural refinement $\rho(M)$. The procedure takes as input a trivially colored Petri net, and a refinement function ρ. It then updates the color sets of the network such that the color set of each place is extended with a new field that will account for the new subtypes of the species that the place stands for. Each transition gets as color set a multiset of the color sets of its pre- and post-places, with multiplicities dictated by the cardinality of each arc expression, just like in the trivial coloring. Note that this means that the refined transition color sets are subtypes of the initial transition color sets, as multisets of subtypes of a color set that is a multiset of supertypes, with identical multiplicities.

Proposition 3. *Given a trivially colored Petri net N that is an implementation of a reaction-based model M, and a full structural refinement relation ρ of M, the colored Petri net $N' = \textsc{TypeRef}(N, \rho)$ is a type refinement of the initial network.*

Algorithm 1 TypeRef

function TYPEREF(N, ρ)
$\Sigma' \leftarrow \varnothing$;
▷ create the new color sets based on the old ones;
for all $p \in P$ **do**
$cs \leftarrow C(p)$;
define a new color set cs' that extends cs with a new field with $\rho(\delta^{-1}(p))$ values;
$\Sigma' \leftarrow \Sigma' \cup \{cs'\}$;
$C'(p) \leftarrow cs'$;
end for
for all $t \in T$ **do**
define cs as a multiset $cs : \{C'(p) \mid p \in P\} \rightarrow \mathbb{N}$ such that $cs(C'(p)) = C(t)(C(p)),\ \forall p \in P$;
$\Sigma' \leftarrow \Sigma' \cup \{cs\}$;
$C'(t) \leftarrow cs$;
end for
▷ re-type the arc expressions: for each variable in an arc expression, create one having as type the new color set of the place that the arc is connected to; the new arc expression is a multiset sum of these variables;
$E' \leftarrow \varnothing$;
for all $e \in E$ **do**
$p \leftarrow$ the place connected to e;
$V \leftarrow$ set of variables appearing in e;
$V' \leftarrow \varnothing$;
for all $v_i \in V$ **do**
define $v_i' : C'(p)$;
$V' \leftarrow V' \cup \{v_i'\}$
end for
$e' \leftarrow {}^{++}\sum_{v \in V'} v$;
$E' \leftarrow E' \cup \{e'\}$;
end for
$\mathbb{M}' \leftarrow \mu\{(p, c) \mid p \in P, c \in C'(p)\}$;
$\mathbb{Y}' \leftarrow \mu\{(t, c) \mid t \in T, c \in C'(t)\}$;
$\mathbb{M}'_0$ is designed such that $\sum_{c \in C'(p)} \mid \mathbb{M}'_0(p, c) \mid = \mid \mathbb{M}_0(p, \mathcal{C}(p)) \mid, \forall p \in P$;
$N' \leftarrow (P, T, A, \Sigma', C', E', \mathbb{M}', \mathbb{Y}', M'_0)$;
return N'
end function

Proof. Based on the construction described in Algorithm 1, we detail here the type refinement morphism between the two networks.

Note that N is trivially colored, so all color sets have exactly one color. The projection from any color in a color set of Σ' onto its corresponding supertype color set is the one color in the supertype color set: $\Pi_{C(x)}(c) = \mathcal{C}(x)$, for any $x \in P \cup T$, and any color $c \in C'(x)$.

We now describe a morphism $\Phi_\rho : N' \to N$ between the two networks, that is a type morphism.

1. $\Phi_\rho(x) = x$ for all $x \in P \cup T \cup A$.

2. $\Phi_\rho(C')(x) = C(x)$. By definition of the color sets in N', the color set of each place and of each transition in N' is a subtype of the color set of the same place/transition in N, i.e. $C'(x) <: \Phi_\rho(C')(x)$. Moreover, for any color $c \in C'(x) : \Pi_{\Phi_\rho(C')(x)}(c) = \Pi_{C(x)}(c) = \mathcal{C}(x)$.

3. $\forall x \in P \cup T : \forall c \in C'(x) : \Phi_\rho(1\,\grave{}\,(x, c)) = 1\,\grave{}\,(x, \Pi_{C(x)}(c)) = 1\,\grave{}\,(x, \mathcal{C}(x))$: for every colored place/transition in N' with color c, the morphism Φ_ρ returns the same place/transition (because Φ_ρ is the identity on $P \cup T$), having as color the projection of c on the color set of x as given by the morphism Φ_ρ, namely $\mathcal{C}(x)$.

4. $\forall (p, t) \in A : \forall (t, m) \in \mathbb{Y}' : \Phi_\rho(E'(p, t)) = E(p, t)$ and the multiset of colored tokens consumed from place p at the firing of transition t in mode m is $E'(p, t)(m)$. By construction of E', the number of consumed tokens is $E(p, t)(\mathcal{C}(t))$. The projection of every color in $C'(p)$ is $\mathcal{C}(p)$, thus we get:

$$\begin{aligned}\Phi_\rho(E^-(1\,\grave{}\,(t, m))(p)) = \Pi_{\Phi_\rho(C')(p)}(E'(p, t)(m)) = E(p, t)(\mathcal{C}(t)) = \\ = E(p, t)(\Pi_{C(t)}(m)) = \Phi_\rho(E')(p, t)(\Pi_{\Phi_\rho(C')(t)}(m)).\end{aligned}$$

5. Similarly, $\forall (t, p) \in A : \forall (t, m) \in \mathbb{Y}' : \Phi_\rho(E'(t, p)) = E(t, p)$ and the multiset of colored tokens added to place p at the firing of transition t in mode m is $E'(t, p)(m)$. By construction of E', the number of produced tokens is $E(t, p)(\mathcal{C}(t))$. The projection of every color in $C'(p)$ is $\mathcal{C}(p)$, thus we get:

$$\begin{aligned}\Phi_\rho(E^+(1\,\grave{}\,(t, m))(p)) = \Pi_{\Phi_\rho(C')(p)}(E'(t, p)(m)) = E(t, p)(\mathcal{C}(t)) = \\ = E(t, p)(\Pi_{C(t)}(m)) = \Phi_\rho(E')(t, p)(\Pi_{\Phi_\rho(C')(t)}(m)).\end{aligned}$$

Because the morphism Φ_ρ respects all conditions for being a type refinement of a Petri net it follows that Algorithm 1 computes a type refinement of its input Petri net.

□

Theorem 1. *Given a reaction-based model $M = (\mathscr{S}, \mathscr{R})$, a structural refinement relation $\rho \subseteq \mathscr{S} \times \mathscr{S}'$, and a colored Petri net $N = (P, T, A, \Sigma, C, E, \mathbb{M}, \mathbb{Y}, M_0)$ that is trivially colored and implements model M with function $\delta : \mathscr{S} \cup \mathscr{R} \to P \cup T$, the colored Petri net* TYPEREF(N, ρ) *implements the full structural ρ-refinement of model M.*

Proof. Let N' denote the refined colored Petri net TYPEREF(N, ρ), and let $M' = (\mathscr{S}', \mathscr{R}')$ denote the full structural ρ-refinement M_ρ. By construction of the refined colored Petri net N' there exists a type morphism between N' and N, as detailed in the proof of Proposition 3.

First, note that N is trivially colored and thus the network is equivalent to its unfolding (see Proposition 1). With a slight abuse of notation, we will use x to denote the unfolded equivalent of a place/transition $x \in P \cup T$, $(x, (c(x)))$.

We show now that the unfolding of N' implements the full structural refinement of M.

Let $N^* = \{P^*, T^*, A^*, f^*, M_0^*\}$ be the unfolding of N'. The color set of a place $p \in P'$ has $\mid \rho(\delta^{-1}(p)) \mid$ elements, where each color represents one refined species $S' \in \mathscr{S}'$, $(\delta^{-1}(p), S') \in \rho$. The places of N^* represent pairs (p, c) such that $p \in P$ and $c \in C'(p)$. Given that every place p has a symbolic correspondence with one species $S = \delta^{-1}(p)$ in $\mathscr{S}$, and the colors of places in N' can be thought of as the refinements of S, there exists a one-to-one correspondence between places in P^* and species in $\mathscr{S}'$. Let $\delta_\rho : \mathscr{S}' \to P^*$, with $\delta_\rho(S') = (\delta(S), c) \in P^*$ where $(S, S') \in \rho$ and no two siblings are mapped to the same value.

δ_ρ can be extended to map also (t, m) pairs to reactions in $\mathscr{R}'$. The color m of a transition t uniquely identifies its pre- and post-places in the unfolded network, and the arc inscriptions. By definition of the color sets of transitions as multisets over the color sets of neighbouring places, it follows that every possible combination of colored tokens flowing through a transition is captured by a transition color. This means that a transition t in N' encodes all possible refinements $\rho(r)$ of the reaction $r = \delta^{-1}(t)$ that transition t stands for in N.

A transition $(t, m) \in T^*$ encodes the reaction

$$\sum_{(p,c) \in {}^\bullet(t,m)} f^*((p,c),(t,m))\delta_\rho^{-1}((p,c)) \to \sum_{(p,c) \in (t,m)^\bullet} f^*((t,m),(p,c))\delta_\rho^{-1}((p,c)). \tag{2}$$

The reaction $r' = \delta_\rho(t)$ that a transition $t \in T$ implements in N' is a ρ-refinement of the reaction $r = \delta(t)$ that the same transition implements in N. This comes from the type refinement conditions 4 and 5 (see Definition 8). The incremental effects of executing a step (t, m) in the refined network equal the incremental effects of executing the step $(t, \Pi_{C(t)}(m))$ in the initial network. The negative incremental effect E^- encodes the left hand side of a reaction, and the positive incremental effect E^+ encodes the right hand side.

We detail here the negative incremental effect of a step, and relate it to its meaning in the model M'. $E^-(1`(t, m)) = \sum_{(p,t) \in A} p \times E((p, t))(m)$. In the

unfolded network N^* a transition (t, m) is connected to places via edges $((p, c), (t, m)) \in A^*$ where $f^*((p, c), (t, m)) = E((p, t))(m)\langle c \rangle$. Summing over all unfolded instances of a place in N^* yields

$$\sum_{c \in C'(p)} f^*((p, c), (t, m)) = \sum_{c \in C'(p)} E((p, t))(m)\langle c \rangle = \mid E((p, t))(m) \mid .$$

Note that the arc expressions in N and N' are the same, which means that their cardinality is also the same. N implements model M, thus $|E((p, t))| = c_{i,j}$ and $|E((t, p))| = c'_{i,j}$ where $c_{i,j}$ is the stoichiometric coefficient of species $S_i = \delta^{-1}(p)$ on the left hand side of reaction $r_j = \delta^{-1}(t)$ and $c'_{i,j}$ is the soichiometric coefficient of S_i on the right hand side of r_j. Arc multiplicities in N^* represent stoichiometries, and for any place p of N' its unfolded places $\{(p, c), \mid \forall c \in C'(p)\}$ represent the sibling species in $\rho(\delta^{-1}(p))$.

A similar argument can be made for the right hand side of a reaction, starting from the positive incremental effect of a step. With both the left and the right hand side of a reaction represented by (t, m) being a ρ-refinement of the left or right, respectively hand side of the reaction $\delta^{-1}(t)$, it follows that (t, m) implements a ρ-refinement of the reaction implemented by t. □

5 Discussion

In this paper we have made a connection between the notions of type refinement of a colored Petri net proposed in [10] and that of full structural refinement of reaction network models proposed in [5]. The connection is based on modeling a reaction network system as a Petri net and using a coloring scheme that allows for easy type refinement. Starting from a Petri net implementation of a reaction-based model, we proposed a general coloring scheme that uses record color sets and further detailed the construction and how the color sets can be refined. We proved that the colored Petri net obtained by coloring the initial Petri net with our coloring strategy is also an implementation of the model implemented by the initial net. We further proved that our strategy is in fact using a type refinement that implements a full structural refinement of a model.

The size of the refined colored Petri net model We discuss here about the size of the colored Petri net model obtained by refining a given model, in terms of number of places and transitions.

A type refinement of a colored Petri net preserves the structure of the network unchanged, i.e. the number of places and transitions does not change. But the semantics of each place and transition is different, and we will therefore consider the unfolding of the colored Petri net.

Given $N = (P, T, A, \Sigma, C, E, \mathbb{M}, \mathbb{Y}, M_0)$ a trivial colored Petri net implementation of a reaction-based model $M = \mathscr{S}, \mathscr{R}$, a refinement relation $\rho \subseteq \mathscr{S} \times \mathscr{S}'$ and a colored Petri net $N' = (P, T, A, \Sigma', C', E, \mathbb{M}', \mathbb{Y}', M_0')$ which is the implementation of the full structural $\rho-$refinement of M by algorithm 1 with function $\delta : \mathscr{S} \cup \mathscr{R} \to P \cup T$, we discuss the size of the unfolding of N', denoted byN^*.

N has by construction $|\mathscr{S}|$ places and $|\mathscr{R}|$ transitions. In N' by construction each place representing a species $S \in \mathscr{S}$ has $\rho(S)$ colors, and will therefore unfold to $\rho(S)$ places. The total number of unfolded places is $\sum_{S \in \mathscr{S}} |\rho(S)| = |\mathscr{S}'|$. The total number of possible colors of a transition depends on the number of colors in the color set of the pre- and post-places of the transition, and on the cardinality of the arc expressions of arcs connected on either end with the transition. A transition $t \in T$ will thus unfold to

$$\prod_{p \in {}^\bullet t} \left\langle\!\!\left\langle \begin{matrix} |\rho(\delta^{-1}(p))| \\ E((p,t)) \end{matrix} \right\rangle\!\!\right\rangle \cdot \prod_{p \in t^\bullet} \left\langle\!\!\left\langle \begin{matrix} |\rho(\delta^{-1}(p))| \\ E, ((t,p)) \end{matrix} \right\rangle\!\!\right\rangle$$

transitions in N^*, which yields a total number of transitions in N^* equal to

$$\sum_{t \in T} \left(\prod_{p \in {}^\bullet t} \left\langle\!\!\left\langle \begin{matrix} |\rho(\delta^{-1}(p))| \\ E((p,t)) \end{matrix} \right\rangle\!\!\right\rangle \cdot \prod_{p \in t^\bullet} \left\langle\!\!\left\langle \begin{matrix} |\rho(\delta^{-1}(p))| \\ E, ((t,p)) \end{matrix} \right\rangle\!\!\right\rangle \right).$$

Depending on the refinement function ρ, this number can be much larger than the number of transitions in the colored network N', which successfully avoids this explosion in number of places and transitions of the network.

Consecutive full structural refinements Very often models go through several steps of refinement, as new information about the modeled system is available, and a more detailed representation is needed. We discuss in this paragraph how subsequent full structural refinements of a model can be implemented using our approach. The problem can be formulated as follows. Given a reaction-based model $M = (\mathscr{S}, \mathscr{R})$ and two refinement relations $\rho \subseteq \mathscr{S} \times \mathscr{S}'$ and $\rho' \subseteq \mathscr{S}' \times \mathscr{S}''$, obtain the full structural $\rho'-$refinement of the full structural $\rho-$refinement of M. In our construction, we start from a trivial coloring of a Petri net implementation of a model. This is however not a limitation of the approach, since subsequent refinements can be implemented as one single refinement that is the composition of the two (or more) successive refinements to be implemented.

We conclude that colored Petri nets can be used to implement full structural refinements of reaction-based models. The major advantage of using the colored Petri nets formalism lies in their ability to represent the fully structurally refined system in a compact way, using the same network structure and adding all refinement details in the colors of places and transitions.

References

[1] Ralph-Johan Back and Joakim Wright. *Refinement calculus: a systematic introduction*. springer Heidelberg, 1998.

[2] MaryAnn Blätke, Christian Rohr, Monika Heiner, and Wolfgang Marwan. A petri-net-based framework for biomodel engineering. In Peter Benner, Rolf Findeisen, Dietrich Flockerzi, Udo Reichl, and Kai Sundmacher, editors, *Large-Scale Networks in Engineering and Life Sciences*, Modeling and Simulation in Science, Engineering and Technology, pages 317–366. Springer International Publishing, 2014.

[3] Christine Choppy, Laure Petrucci, and Alfred Sanogo. Coloured petri nets refinements. In *PNSE+ ModPE*, pages 187–201. Citeseer, 2013.

[4] V. Danos, J. Feret, W. Fontana, R. Harmer, and J. Krivine. Rule-based modelling and model perturbation. *Transactions on Computational Systems Biology XI*, pages 116–137, 2009.

[5] Cristian Gratie and Ion Petre. Fit-preserving data refinement of mass-action reaction networks. In Arnold Beckmann, Erzsébet Csuhaj-Varjú, and Klaus Meer, editors, *Language, Life, Limits*, volume 8493 of *Lecture Notes in Computer Science*, pages 204–213. Springer, 2014.

[6] Bogdan Iancu, Elena Czeizler, Eugen Czeizler, and Ion Petre. Quantitative refinement of reaction models. *International Journal of Unconventional Computing*, 8(5-6):529–550, 2012.

[7] Bogdan Iancu, Diana-Elena Gratie, Sepinoud Azimi, and Ion Petre. On the implementation of quantitative model refinement. In Adrian-Horia Dediu, Carlos Martín-Vide, and Bianca Truthe, editors, *Algorithms for Computational Biology*, volume 8542 of *Lecture Notes in Computer Science*, pages 95–106. Springer International Publishing, 2014.

[8] Kurt Jensen. Coloured petri nets: A high level language for system design and analysis. *Advances in Petri nets 1990*, pages 342–416, 1991.

[9] Kurt Jensen. Coloured petri nets. volume 1, basic concepts. *EATCS Monographs in Theoretical Computer Science*, 1, 1992.

[10] Charles Lakos. Composing abstractions of coloured petri nets. In *Application and Theory of Petri Nets 2000*, pages 323–342. Springer, 2000.

[11] Charles Lakos and Glenn Lewis. A catalogue of incremental changes for coloured petri nets. Technical report, the International Conference of Application and Theory of Petri Nets, 1999.

[12] Andrzej Mizera, Eugen Czeizler, and Ion Petre. Self-assembly models of variable resolution. *LNBI Transactions on Computational Systems Biology*, pages 181–203, 2011.

[13] Tadao Murata. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77(4):541–580, 1989.

[14] E. Murphy, V. Danos, J. Feret, J. Krivine, and R. 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.

[15] Marco Pistore and Davide Sangiorgi. A partition refinement algorithm for the pi-calculus. In *Proceedings of CAV'96, volume 1102 of Lecture Notes in Computer Science*, pages 200–1. Springer-Verlag, 1996.

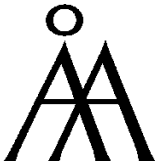
[16] Ichiro Suzuki and Tadao Murata. A method for stepwise refinement and abstraction of petri nets. *Journal of Computer and System Sciences*, 27(1):51 – 76, 1983.

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

Turku School of Economics
- Institute of Information Systems Sciences



Åbo Akademi University
- Department of Computer Science
- Institute for Advanced Management Systems Research

ISBN 978-952-12-3209-1
ISSN 1239-1891