Reconstructing $\mathcal{X}'$-deterministic extended Petri nets from experimental time-series data $\mathcal{X}'$

Marie C.F. Favre*, Annegret K. Wagler

Laboratoire d'Informatique, de Modélisation et d'Optimisation des Systèmes (LIMOS, UMR CNRS 6158), Université Blaise Pascal (Clermont-Ferrand II) BP 10125, 63173 Aubière Cedex, France
marie.favre@isima.fr wagler@isima.fr

Abstract. This work aims at reconstructing Petri net models for biological systems from experimental time-series data $\mathcal{X}'$. The reconstructed models shall reproduce the experimentally observed dynamic behavior in a simulation. For that, we consider Petri nets with priority relations among the transitions and control-arcs, to obtain additional activation rules for transitions to control the dynamic behavior. The contribution of this paper is to present an integrative reconstruction method, taking both concepts, priority relations and control-arcs, into account. Our approach is based on previous works for special cases and shows how these known steps have to be modified and combined to generate the desired integrative models, called $\mathcal{X}'$-deterministic extended Petri nets.

1 Introduction

The overall aim of systems biology is to analyze biological systems and to understand different phenomena therein as, e.g., responses of cells to environmental changes, host-pathogen interactions, or effects of gene defects. To gain the required insight into the underlying biological processes, experiments are performed and the resulting experimental data are interpreted in terms of models. Depending on the biological aim and the type and quality of the available data, different types of mathematical models are used and corresponding methods for their reconstruction have been developed. Our work is dedicated to Petri nets, a framework which turned out to coherently model static interactions in terms of networks and dynamic processes in terms of state changes, see e.g. [5,9]. A network $(P, T, \mathcal{A}, w)$ reflects the involved system components by places $p \in P$ and their interactions by transitions $t \in T$, linked by weighted directed arcs. Each place $p \in P$ can be marked with an integral number of tokens defining a system state $\boldsymbol{x} \in \mathbb{Z}_+^{|P|}$, dynamic processes are represented by sequences of state changes, performed by switching or firing enabled transitions (see Section 2).

* This work was founded by the French National Research Agency, the European Commission (Feder funds) and the Région Auvergne in the Framework of the LabEx IMobS[3].

G. Balbo and M. Heiner (Eds.): BioPPN 2013, a satellite event of PETRI NETS 2013,
CEUR Workshop Proceedings Vol. 988, 2013.

Our central question is to reconstruct models of this type from experimental time-series data by means of an exact, exclusively data-driven approach. We base our method on earlier results from [1,2,3,4,8,12]. This approach takes as input a set P of places and discrete time-series data $\mathcal{X}'$ given by sequences $(\boldsymbol{x}^0, \boldsymbol{x}^1, \ldots, \boldsymbol{x}^k)$ of experimentally observed system states. The goal is to determine all Petri nets $(P, T, \mathcal{A}, w)$ that are able to reproduce the data, i.e., that perform for each $\boldsymbol{x}^j \in \mathcal{X}'$ the experimentally observed state change to $\boldsymbol{x}^{j+1} \in \mathcal{X}'$ in a simulation. Hence, in contrast to the normally used stochastic simulation, we require that for states where at least two transitions are enabled, the decision between the different alternatives is not taken randomly, but a specific transition is selected. For that, (standard) Petri nets have to be equipped with additional activation rules to force the switching or firing of special transitions (to reach $\boldsymbol{x}^{j+1}$ from $\boldsymbol{x}^j$), and to prevent all others from switching. Analogously, the reconstruction approach needs to be extended accordingly. In previous works, we considered two possible types of additional activation rules.

On the one hand, in [8,11,12] the concept of priority relations among the transitions of a network was introduced in order to allow the modelization of deterministic systems (see Section 2 for more details). This leads to the notion of $\mathcal{X}'$-deterministic Petri nets, which show a prescribed behavior on the experimentally observed subset $\mathcal{X}'$ of states: the reconstructed Petri nets $(P, T, \mathcal{A}, w)$ do not only contain enough transitions to reach the experimentally observed successors $\boldsymbol{x}^{j+1}$ from $\boldsymbol{x}^j$, but exactly this transition will be selected among all enabled ones in $\boldsymbol{x}^j$ which is necessary to reach $\boldsymbol{x}^{j+1}$.

On the other hand, in [1,2] the concept of control-arcs was used to represent catalytic or inhibitory dependencies. Here, an enabled transition $t \in T$ coupled with a read-arc (resp. an inhibitory-arc) to a place $p \in P$ can switch only if a token (resp. no token) is present in p (see Section 2). This leads to the reconstruction of extended Petri nets which are catalytic conformal with $\mathcal{X}'$.

For consistently integrating both concepts, priority relations and control-arcs, into the modeling framework, the difficulty is that both are concurrent concepts to force or prevent the switching of enabled transitions. In [13], the notion of $\mathcal{X}'$-deterministic extended Petri nets is introduced as the desired output of an integrative reconstruction method. The contribution of this paper is to present the steps of such an approach, based on previous reconstruction methods for special cases [1,2,3,4,8] , and to show how these known steps have to be modified and combined to generate the desired integrative models (see Section 3).

2 Petri nets and extensions

A standard or simple *Petri net* $\mathcal{P} = (P, T, A, w)$ is a weighted directed bipartite graph with two kinds of nodes, places and transitions. The places $p \in P$ represent the system components (e.g. proteins, enzymes, genes, receptors or their conformational states) and the transitions $t \in T$ stand for their interactions (e.g., chemical reactions, activations or causal dependencies). The arcs in

$A \subset (P \times T) \cup (T \times P)$ link places and transitions, and the arc weights $w : A \to \mathbb{N}$ reflect stoichiometric coefficients of the corresponding reactions.

Each place $p \in P$ can be marked with an integral number x_p of tokens, and any marking defines a state $\boldsymbol{x} \in \mathbb{N}^{|P|}$ of the system. In biological systems, all components can be considered to be bounded, as the value x_p of any state refers to the concentration of the studied component $p \in P$, which can only increase up to a certain maximum $\mathrm{cap}(p)$. This leads to a *capacitated Petri net* $(\mathcal{P}, \mathrm{cap})$, i.e., a Petri net $\mathcal{P} = (P, T, A, w)$ together with a capacity function $\mathrm{cap} : P \to \mathbb{N}$, whose set of potential states is $\mathcal{X} := \{\boldsymbol{x} \in \mathbb{N}^{|P|} \mid x_p \leq \mathrm{cap}(p)\}$. A transition $t \in T$ is *enabled* in a state $\boldsymbol{x} \in \mathcal{X}$ of a capacitated Petri net if

E1 $x_p \geq w(p,t)$ for all p with $(p,t) \in A$, and,
E2 $x_p + w(t,p) \leq \mathrm{cap}(p)$ for all p with $(t,p) \in A$

and we define $T(\boldsymbol{x}) := \{t \in T : t \text{ satisfies E1, E2 in } \boldsymbol{x}\}$.
An *extended Petri net* $\mathcal{P} = (P, T, (A \cup A_R \cup A_I), w)$ is a Petri net which has, besides the (standard) arcs in A, two additional sets of so-called control-arcs: the set of read-arcs $A_R \subset P \times T$ and the set of inhibitor-arcs $A_I \subset P \times T$. We denote the set of control-arcs by $A_C = A_R \cup A_I$, and the set of all arcs by $\mathcal{A} = A \cup A_R \cup A_I$.

In a capacitated extended Petri net, switching of transitions is additionally controlled by read- and inhibitor-arcs; a transition t satisfying E1 and E2 can switch only if also the following conditions hold:

E3 $x_p \geq w(p,t)$ for all p with $(p,t) \in A_R$, and,
E4 $x_p < w(p,t)$ for all p with $(p,t) \in A_I$.

In an extended Petri net, a transition is *enabled* in a state $x \in \mathcal{X}$ if it satisfies E1, ... , E4 (otherwise, it is *disabled*). The switch of a transition t enabled in x leads to a successor state $\mathrm{succ}_{\mathcal{X}}(\boldsymbol{x}) = \boldsymbol{x}' \in \mathcal{X}$ whose marking is obtained by

$$x'_p := \begin{cases} x_p - w(p,t), & \text{for all } p \text{ with } (p,t) \in A, \\ x_p + w(t,p), & \text{for all } p \text{ with } (t,p) \in A, \\ x_p, & \text{otherwise.} \end{cases}$$

In general, there can be more than one transition satisfying E1, ... , E4 in a state $\boldsymbol{x} \in \mathcal{X}$ and we define $T_{\mathcal{A}}(\boldsymbol{x}) := \{t \in T : t \text{ satisfies E1, \ldots, E4 in } \boldsymbol{x}\}$. The decision which transition switches is typically taken randomly (and the dynamic behavior is analyzed in terms of reachability, starting from a certain initial state). This is not appropriate for modeling biological systems which show a deterministic behavior, e.g., where a certain stimulation always results in the same response. In this case, additional activation rules are required in order to force the switch from a state $\boldsymbol{x}$ to a specific successor state $\mathrm{succ}_{\mathcal{X}}(\boldsymbol{x})$. For this purpose, priorities between the transitions of the network can be used to determine which of the transitions in $T_{\mathcal{A}}(\boldsymbol{x})$ has to be taken. Note that these priorities typically reflect the rate of the corresponding reactions where the fastest reaction has highest priority. In Marwan et al. [8] it is proposed to model such priorities with the help of partial orders on the set T of transitions of the network $\mathcal{P}$. Here, a *partial order* $\mathcal{O}$ on T is a relation $\leq$ between pairs of elements of T respecting

– reflexivity (i.e., $t \leq t$ holds for all $t \in T$),
– transitivity (i.e., from $t \leq t'$ and $t' \leq t''$ follows $t \leq t''$ for all $t, t', t'' \in T$),
– anti-symmetry (i.e., $t \leq t'$ and $t' \leq t$ implies $t = t'$).

We call $(\mathcal{P}, \mathcal{O})$ an *(extended) Petri net with priorities*, if $\mathcal{P} = (P, T, \mathcal{A}, w)$ is an (extended) Petri net and $\mathcal{O}$ a priority relation on T.

Note that priorities can prevent enabled transitions from switching: for a state $\boldsymbol{x} \in \mathcal{X}$, only a transition $t \in T_{\mathcal{A}}(\boldsymbol{x})$ is *allowed to switch* or *can switch* if

E5 there is no other transition $t' \in T_{\mathcal{A}}(\boldsymbol{x})$ with $(t \leq t') \in \mathcal{O}$.

The set of all transitions that are allowed to switch in $\boldsymbol{x}$ is denoted by

$$T_{\mathcal{A},\mathcal{O}}(\boldsymbol{x}) := \{t \in T : t \text{ satisfies E1, } \dots \text{, E5 in } \boldsymbol{x}\}.$$

To enforce a deterministic behavior, $T_{\mathcal{A},\mathcal{O}}(\boldsymbol{x})$ must contain at most one element for each $\boldsymbol{x} \in \mathcal{X}$ to enforce that $\boldsymbol{x}$ has a unique successor $\mathrm{succ}_{\mathcal{X}}(\boldsymbol{x})$, see [11] for more details. Extended Petri nets with priorities satisfying this property are said to be $\mathcal{X}$-deterministic. For our purpose, we consider a relaxed condition, namely that $T_{\mathcal{A},\mathcal{O}}(\boldsymbol{x})$ contains at most one element for each experimentally observed state $\boldsymbol{x} \in \mathcal{X}'$, but $T_{\mathcal{A},\mathcal{O}}(\boldsymbol{x})$ may contain several elements for non-observed states $\boldsymbol{x} \in \mathcal{X} \setminus \mathcal{X}'$. We call such Petri nets *$\mathcal{X}'$-deterministic.*

In this paper we consider *capacitated extended Petri nets with priorities* $(\mathcal{P}, \mathrm{cap}, \mathcal{O})$: extended Petri nets $\mathcal{P} = (P, T, \mathcal{A}, w)$ with a capacity function $\mathrm{cap} : P \to \mathbb{N}$ on their places and a partial order $\mathcal{O} \subset T \times T$ on their transitions. Our goal is to reconstruct $\mathcal{X}'$-deterministic extended Petri nets from given experimental data $\mathcal{X}'$.

3 Reconstructing $\mathcal{X}'$-deterministic extended Petri nets

In this section, we describe the input, the main ideas, and the generated output of our integrative reconstruction approach.

3.1 Input

A set of components P (later represented by the set of places) is chosen which is expected to be crucial for the studied phenomenon. All known P-invariants [1] of the system (e.g., different conformational stages of a cell, a receptor, a protein) shall be collected in a set $\mathcal{I}_P$.

To perform an experiment, one first triggeres the system in some state $\boldsymbol{x^0}$ (by external stimuli like the change of nutrient concentrations or the exposition to some pathogens), to generate an initial state $\boldsymbol{x^1}$. Then the system's response to the stimulation is observed and the resulting state changes are measured

[1] Laxly said, a P-invariant is a set $P' \subseteq P$ of places (components) where the sum of the number of all tokens on all the places in P' is constant. P-invariants are not computed by the algorithm but must be known a priori by a biologist.

for all components at certain time points. This yields a sequence of (discrete or discretized) states $\boldsymbol{x}^j \in \mathbb{Z}^{|P|}$ reflecting the time-dependent response of the system to the stimulation in $\boldsymbol{x}^1$, which typically terminates in a terminal state $\boldsymbol{x}^k$ where no further changes are observed. The corresponding experiment is

$$\mathcal{X}'(\boldsymbol{x}^1, \boldsymbol{x}^k) = (\boldsymbol{x}^0; \boldsymbol{x}^1, \ldots, \boldsymbol{x}^k).$$

Several experiments starting from different initial states in a set $\mathcal{X}'_{ini} \subseteq \mathcal{X}'$, reporting the observed state changes for all components $p \in P$ at certain time points, and ending at different terminal states in a set $\mathcal{X}'_{term} \subseteq \mathcal{X}'$ describe the studied phenomenon, and yield experimental time-series data of the form

$$\mathcal{X}' = \{\mathcal{X}'(\boldsymbol{x}^1, \boldsymbol{x}^k) : \boldsymbol{x}^1 \in \mathcal{X}'_{ini}, \boldsymbol{x}^k \in \mathcal{X}'_{term}\}.$$

Thus, the input of the reconstruction approach is given by $(P, \mathcal{I}_P, \mathcal{X}')$.

Example 1. As running example, we will consider experimental biological data from the *light-induced sporulation of Physarum polycephalum*. The developmental decision of starving *P. polycephalum* plasmodia to exit the vegetative plasmodial stage and to enter the sporulation pathway is controlled by environmental factors like visible light [10]. One of the photoreceptors involved in the control of sporulation *Spo* is a phytochrome-like photoreversible photoreceptor protein which occurs in two stages P_{FR} and P_R. If the dark-adapted form P_{FR} absorbs far-red light FR, the receptor is converted into its red-absorbing form P_R, which causes sporulation [6]. If P_R is exposed to red light R, it is photoconverted back to the initial stage P_{FR}, which prevents sporulation. Note that the changes between the stages P_{FR} and P_R can be experimentally observed due to a change of color. The experimental setting consists of

$$\begin{array}{lll} P = \{FR, R, P_{FR}, P_R, Spo\} & \mathcal{X}'(\boldsymbol{x}^1, \boldsymbol{x}^3) = (\boldsymbol{x}^0; \boldsymbol{x}^1, \boldsymbol{x}^2, \boldsymbol{x}^3) & \mathcal{X}'_{ini} = \{\boldsymbol{x}^1, \boldsymbol{x}^4\} \\ \mathcal{I}_P = \{P_{FR}, P_R\} & \mathcal{X}'(\boldsymbol{x}^4, \boldsymbol{x}^0) = (\boldsymbol{x}^2; \boldsymbol{x}^4, \boldsymbol{x}^0) & \mathcal{X}'_{term} = \{\boldsymbol{x}^3, \boldsymbol{x}^0\} \end{array}$$

as input for the algorithm, we represent all observed states schematically in Fig 1.

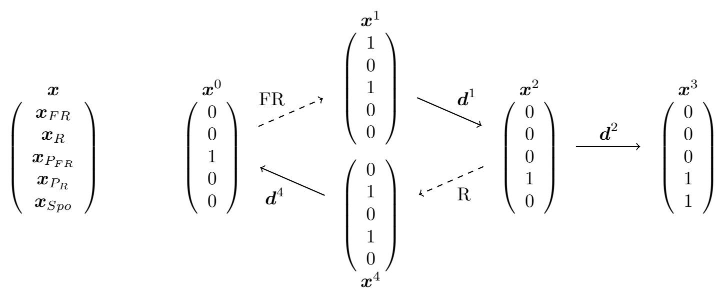

Fig. 1. A scheme illustrating the experimental time-series data described in Exp. 1 concerning the light-induced sporulation of *Physarum polycephalum*, where the entries of the state vectors are interpreted as shown on the left (dashed arrows represent stimulations, solid arrows responses).

In the best case, two consecutively measured states $\boldsymbol{x}^j, \boldsymbol{x}^{j+1} \in \mathcal{X}'$ are also consecutive system states, i.e., $\boldsymbol{x}^{j+1}$ can be obtained from $\boldsymbol{x}^j$ by switching a single transition in T. This is, however, in general not the case (and depends on the chosen time points to measure the states in $\mathcal{X}'$), but $\boldsymbol{x}^{j+1}$ is obtained from $\boldsymbol{x}^j$ by a switching sequence of some length, where the intermediate states are not reported in $\mathcal{X}'$.

For a successful reconstruction approach, the data $\mathcal{X}'$ need to satisfy two properties: reproducibility and monotonicity. The data $\mathcal{X}'$ are *reproducible* if for each $\boldsymbol{x}^j \in \mathcal{X}'$ there is a unique observed successor state $\mathrm{succ}_{\mathcal{X}'}(\boldsymbol{x}^j) = \boldsymbol{x}^{j+1} \in \mathcal{X}'$. Moreover, the data $\mathcal{X}'$ are *monotone* if for each pair $(\boldsymbol{x}^j, \boldsymbol{x}^{j+1}) \in \mathcal{X}'$, the possible intermediate states $\boldsymbol{x}^j = \boldsymbol{y}^1, \boldsymbol{y}^2, ..., \boldsymbol{y}^{m+1} = \boldsymbol{x}^{j+1}$ satisfy

$y_p^1 \leq y_p^2 \leq \ldots \leq y_p^m \leq y_p^{m+1}$ for all $p \in P$ with $x_p^j \leq x_p^{j+1}$ and
$y_p^1 \geq y_p^2 \geq \ldots \geq y_p^m \geq y_p^{m+1}$ for all $p \in P$ with $x_p^j \geq x_p^{j+1}$.

Whereas reproducibility is obviously necessary, it was shown in [3] that monotonicity [2] has to be required too. Due to monotonicity, a capacity $\mathrm{cap}(p)$ can be determined from $\mathcal{X}'$ for each $p \in P$ by $\mathrm{cap}(p) = \max\{x_p : \boldsymbol{x} \in \mathcal{X}'\}$, but is not required for the reconstruction.

3.2 Output

A capacitated extended Petri net with priorities $(\mathcal{P}, \mathrm{cap}, \mathcal{O})$ with $\mathcal{P} = (P, T, \mathcal{A}, w)$ fits the given data $\mathcal{X}'$ when it is able to perform every observed state change from $\boldsymbol{x}^j \in \mathcal{X}'$ to $\mathrm{succ}_{\mathcal{X}'}(\boldsymbol{x}^j) = \boldsymbol{x}^{j+1} \in \mathcal{X}'$. This can be interpreted as follows. With $\mathcal{P}$, an *incidence matrix* $M(\mathcal{P}) \in \mathbb{Z}^{|P| \times |T|}$ is associated, where each row corresponds to a place $p \in P$ of the network, and each column $M(\mathcal{P})_{\cdot t}$ to the *update vector* $\boldsymbol{r}^t$ of a transition $t \in T$:

$$r_p^t = M(\mathcal{P})_{pt} := \begin{cases} -w(p,t) & \text{if } (p,t) \in A, \\ +w(t,p) & \text{if } (t,p) \in A, \\ 0 & \text{otherwise.} \end{cases}$$

Reaching $\boldsymbol{x}^{j+1}$ from $\boldsymbol{x}^j$ by a switching sequence using the transitions from a subset $T' \subseteq T$ is equivalent to obtain the state vector $\boldsymbol{x}^{j+1}$ from $\boldsymbol{x}^j$ by adding the corresponding columns $M(\mathcal{P})_{\cdot t}$ of $M(\mathcal{P})$ for all $t \in T'$:

$$\boldsymbol{x}^j + \sum_{t \in T'} M(\mathcal{P})_{\cdot t} = \boldsymbol{x}^{j+1}.$$

Hence, T has to contain enough transitions to perform all experimentally observed switching sequences. The underlying standard network $\mathcal{P} = (P, T, A, w)$ is

[2] This is equivalent to say that system states in $\mathcal{X}'$ have been measured to appropriate time points such that the values of their components do not oscillate between two measured states or, equivalently, that all essential responses are indeed reported in the experiments.

conformal with $\mathcal{X}'$ if, for any two consecutive states $\boldsymbol{x}^j, \mathrm{succ}_{\mathcal{X}'}(\boldsymbol{x}^j) = \boldsymbol{x}^{j+1} \in \mathcal{X}'$, the linear equation system

$$\boldsymbol{x}^{j+1} - \boldsymbol{x}^j = M(\mathcal{P})\boldsymbol{\lambda}$$

has an integral solution $\boldsymbol{\lambda} \in \mathbb{N}^{|T|}$ such that $\boldsymbol{\lambda}$ is the incidence vector of a sequence $(t^1, ..., t^m)$ of transition switches, i.e., there are intermediate states $\boldsymbol{x}^j = \boldsymbol{y}^1, \boldsymbol{y}^2, ..., \boldsymbol{y}^{m+1} = \boldsymbol{x}^{j+1}$ with $\boldsymbol{y}^l + M(\mathcal{P})_{\cdot t^l} = \boldsymbol{y}^{l+1}$ for $1 \leq l \leq m$.

The extended Petri net $\mathcal{P} = (P, T, \mathcal{A}, w)$ is *catalytic conformal* with $\mathcal{X}'$ if $t^l \in T_{\mathcal{A}}(\boldsymbol{y}^l)$ for each intermediate state $\boldsymbol{y}^l$, and the extended Petri net with priorities $(\mathcal{P}, \mathcal{O})$ is $\mathcal{X}'$*-deterministic* if $\{t^l\} = T_{\mathcal{A},\mathcal{O}}(\boldsymbol{y}^l)$ holds for all $\boldsymbol{y}^l$.

The desired output of the reconstruction approach consists of the set of all $\mathcal{X}'$-deterministic extended Petri nets $(\mathcal{P}, \mathrm{cap}, \mathcal{O})$ (all having the same set P of places and the same capacities cap deduced from $\mathcal{X}'$).

3.3 Representation of the observed responses

To solve the problem of representing the observed responses by switching sequences, we propose the following approach, based on previous works in [4,8].

Extraction of difference vectors. As initial step, extract the observed changes of states from the experimental data. For that, define the set

$$\mathcal{D} := \left\{ \boldsymbol{d}^j = \boldsymbol{x}^{j+1} - \boldsymbol{x}^j \ : \ \boldsymbol{x}^{j+1} = \mathrm{succ}_{\mathcal{X}'}(\boldsymbol{x}^j) \in \mathcal{X}' \right\}.$$

Example 2. From our running example in Fig. 1 we obtain $\mathcal{D} = \{d^1, d^2, d^4\}$ with $\boldsymbol{d}^1 = \boldsymbol{x}^2 - \boldsymbol{x}^1 = (-1, 0, -1, 1, 0)^T$, $\boldsymbol{d}^2 = \boldsymbol{x}^3 - \boldsymbol{x}^2 = (0, 0, 0, 0, 1)^T$ and $\boldsymbol{d}^4 = \boldsymbol{x}^0 - \boldsymbol{x}^4 = (0, -1, 1, -1, 0)^T$.

Generating the complete list of all $\mathcal{X}'$-deterministic extended Petri nets $\mathcal{P} = (P, T, \mathcal{A}, w)$ includes finding the corresponding standard networks and their incidence matrices $M \in \mathbb{Z}^{|P| \times |T|}$.

The first step is to describe the set of potential update vectors which might constitute the columns of M.

Representation of difference vectors. Recall that two consecutively measured states $\boldsymbol{x}^j, \boldsymbol{x}^{j+1} \in \mathcal{X}'$ are not necessarily consecutive system states, i.e., $\boldsymbol{x}^{j+1}$ may be obtained from $\boldsymbol{x}^j$ by a switching sequence of some length, where the intermediate states are not reported in $\mathcal{X}'$. Due to monotonicity, the values of the elements cannot oscillate in the intermediate states between $\boldsymbol{x}^j$ and $\boldsymbol{x}^{j+1}$.

Moreover, for any P-invariant $P' \in \mathcal{I}_P$, all suitable update vectors have to satisfy $\sum_{p \in P'} r_p = 0$. Hence, it suffices to represent any $\boldsymbol{d}^j \in \mathcal{D}$ using only vectors from the following set

$$\mathrm{Box}(\boldsymbol{d}^j) = \left\{ \boldsymbol{r} \in \mathbb{Z}^{|P|} : \begin{array}{r} 0 \leq r_p \leq d_p^j \text{ if } d_p^j > 0 \\ d_p^j \leq r_p \leq 0 \text{ if } d_p^j < 0 \\ r_p = 0 \text{ if } d_p^j = 0 \\ \sum_{p \in P'} r_p = 0 \quad \forall P' \in \mathcal{I}_P \end{array} \right\} \setminus \{\boldsymbol{0}\}.$$

Remark 1. In previous approaches [4], none of the reconstructed (standard) networks must contain a transition enabled at any of the observed terminal states $\boldsymbol{x}^k \in \mathcal{X}'_{term}$; hence all such vectors in $\text{Box}(\boldsymbol{d}^j)$ could be removed. This is not the case for extended Petri nets as desired output of the reconstruction, since the corresponding transitions can be disabled due to control-arcs. Here, we only exclude the zero vector $\mathbf{0}$ as trivial update vector.

Next, we determine for any $\boldsymbol{d}^j \in \mathcal{D}$, the set $\Lambda(\boldsymbol{d}^j)$ of all integral solutions of the equation system

$$\boldsymbol{d}^j = \sum_{\boldsymbol{r}^t \in \text{ Box}(\boldsymbol{d}^j)} \lambda_t \boldsymbol{r}^t, \ \lambda_t \in \mathbb{Z}_+.$$

By construction, $\text{Box}(\boldsymbol{d}^j)$ and $\Lambda(\boldsymbol{d}^j)$ are always non-empty since $\boldsymbol{d}^j$ itself is always a solution due to the required reproducibility of the input data $\mathcal{X}'$ (which particularly includes $\boldsymbol{d}^j \neq \mathbf{0}$ for all $\boldsymbol{d}^j \in \mathcal{D}$). For each $\boldsymbol{\lambda} \in \Lambda(\boldsymbol{d}^j)$, construct the (multi-)set

$$\mathcal{R}(\boldsymbol{d}^j, \boldsymbol{\lambda}) = \{r^t \in \text{Box}(\boldsymbol{d}^j) : \lambda_t \neq 0\}$$

of update vectors used for this solution $\boldsymbol{\lambda}$.

Example 3. For $\mathcal{D}$ from Exp. 2, the update vectors for a decomposition are $\text{Box}(\boldsymbol{d}^1) = \{\boldsymbol{d}^1, \boldsymbol{r}^1, \boldsymbol{r}^2\}$, $\text{Box}(\boldsymbol{d}^2) = \{\boldsymbol{d}^2\}$ and $\text{Box}(\boldsymbol{d}^4) = \{\boldsymbol{d}^4, \boldsymbol{r}^3, \boldsymbol{r}^4\}$ with vectors $\boldsymbol{r}^1 = (-1,0,0,0,0)^T$, $\boldsymbol{r}^2 = (0,0,-1,1,0)^T$, $\boldsymbol{r}^3 = (0,-1,0,0,0)^T$ and $\boldsymbol{r}^4 = (0,0,1,-1,0)^T$. Hence, the possible decomposition of the responses are $\boldsymbol{d}^1 = \boldsymbol{d}^1 = \boldsymbol{r}^1 + \boldsymbol{r}^2$, $\boldsymbol{d}^2 = \boldsymbol{d}^2$ and $\boldsymbol{d}^4 = \boldsymbol{d}^4 = \boldsymbol{r}^3 + \boldsymbol{r}^4$ and the resulting sets are

$$\begin{aligned}
&\mathcal{R}(\boldsymbol{d}^1, \boldsymbol{\lambda}^1) = \{\boldsymbol{d}^1\}, \mathcal{R}(\boldsymbol{d}^1, \boldsymbol{\lambda}^2) = \{\boldsymbol{r}^1, \boldsymbol{r}^2\},\\
&\mathcal{R}(\boldsymbol{d}^2, \boldsymbol{\lambda}) = \{\boldsymbol{d}^2\},\\
&\mathcal{R}(\boldsymbol{d}^4, \boldsymbol{\lambda}^1) = \{\boldsymbol{d}^4\}, \mathcal{R}(\boldsymbol{d}^4, \boldsymbol{\lambda}^2) = \{\boldsymbol{r}^3, \boldsymbol{r}^4\}.
\end{aligned}$$

3.4 Priority conflicts.

To compose all possible standard networks, we have to select exactly one solution $\boldsymbol{\lambda} \in \Lambda(\boldsymbol{d}^j)$ for each $\boldsymbol{d}^j \in \mathcal{D}$ and to take the union of the corresponding sets $\mathcal{R}(\boldsymbol{d}^j, \boldsymbol{\lambda})$ in order to yield the columns $M_{\cdot t} = \boldsymbol{r}^t$ of an incidence matrix M of a conformal network. To ensure that the generated conformal networks can be made $\mathcal{X}'$-deterministic, we proceed as follows.

Sequences and their conflicts. Every permutation $\pi = (\boldsymbol{r}^{t_1}, \ldots, \boldsymbol{r}^{t_m})$ of the elements of a set $\mathcal{R}(\boldsymbol{d}^j, \boldsymbol{\lambda})$ gives rise to a sequence of intermediate states $\boldsymbol{x}^j = \boldsymbol{y}^1, \boldsymbol{y}^2, ..., \boldsymbol{y}^m, \boldsymbol{y}^{m+1} = \boldsymbol{x}^{j+1}$ with

$$\sigma_{\pi,\boldsymbol{\lambda}}(\boldsymbol{x}^j, \boldsymbol{d}^j) = \left((\boldsymbol{y}^1, \boldsymbol{r}^{t_1}), (\boldsymbol{y}^2, \boldsymbol{r}^{t_2}), \ldots, (\boldsymbol{y}^m, \boldsymbol{r}^{t_m})\right).$$

By construction, every such sequence σ respects monotonicity and induces a priority relation $\mathcal{O}_\sigma$, since it implies which transition t^i is supposed to have highest priority (and thus switches) for every intermediate state $\boldsymbol{y}^i$.

To impose valid priority relations $\mathcal{O}$ among all transitions of the reconstructed networks, we have to take priority conflicts between priority relations $\mathcal{O}_\sigma$ induced by different sequences σ into account.

Two sequences σ and σ' are in *priority conflict* if there are update vectors $\boldsymbol{r}^t \neq \boldsymbol{r}^{t'}$ and intermediate states $\boldsymbol{y}, \boldsymbol{y}'$ such that $t, t' \in T(\boldsymbol{y}) \cap T(\boldsymbol{y}')$ and $(\boldsymbol{y}, \boldsymbol{r}^t) \in \sigma$ but $(\boldsymbol{y}', \boldsymbol{r}^{t'}) \in \sigma'$ (since this implies $t > t'$ in $\mathcal{O}_\sigma$ but $t' > t$ in $\mathcal{O}_{\sigma'}$).

We have a *weak priority conflict* (WPC) if $y \neq y'$ and a *strong priority conflict* (SPC) if $y = y'$. Note that a WPC can be resolved by adding appropriate control-arcs, whereas a SPC cannot be resolved that way (see section 3.5).

Note that we have a strong priority conflict between the trivial sequence $\sigma(\boldsymbol{x}^k, \boldsymbol{0})$ for any terminal state $\boldsymbol{x}^k \in \mathcal{X}'_{term}$ and any sequence σ containing $\boldsymbol{x}^k$ as intermediate state. Such sequences σ are not catalytic conformal due to [2].

Example 4. From the running example, we obtain the following sequences

$$\begin{array}{rlrl}
\sigma_1(\boldsymbol{x}^1, \boldsymbol{d}^1) &= ((\boldsymbol{x}^1, \boldsymbol{d}^1)) & \sigma_1(\boldsymbol{x}^4, \boldsymbol{d}^4) &= ((\boldsymbol{x}^4, \boldsymbol{d}^4)) \\
\sigma_2(\boldsymbol{x}^1, \boldsymbol{d}^1) &= ((\boldsymbol{x}^1, \boldsymbol{r}^1), (\boldsymbol{x}^0, \boldsymbol{r}^2)) & \sigma_2(\boldsymbol{x}^4, \boldsymbol{d}^4) &= ((\boldsymbol{x}^4, \boldsymbol{r}^3), (\boldsymbol{x}^2, \boldsymbol{r}^4)) \\
\sigma_3(\boldsymbol{x}^1, \boldsymbol{d}^1) &= ((\boldsymbol{x}^1, \boldsymbol{r}^2), (\boldsymbol{x}^5, \boldsymbol{r}^1)) & \sigma_3(\boldsymbol{x}^4, \boldsymbol{d}^4) &= ((\boldsymbol{x}^4, \boldsymbol{r}^4), (\boldsymbol{x}^6, \boldsymbol{r}^3)) \\
\sigma(\boldsymbol{x}^2, \boldsymbol{d}^2) &= ((\boldsymbol{x}^2, \boldsymbol{d}^2)) & \sigma(\boldsymbol{x}^3, 0) &\text{and } \sigma(\boldsymbol{x}^0, 0)
\end{array}$$

where $\boldsymbol{x}^5 = (1, 0, 0, 1, 0)^T$ and $\boldsymbol{x}^6 = (0, 1, 1, 0, 0)^T$. Between these sequences, we have SPCs and WPCs as indicated in Fig. 2.

Construction of the priority conflict graph. To reflect the weak and strong priority conflicts between all possible sequences resulting from $\mathcal{X}'$, we construct a *priority conflict graph* $\mathcal{G} = (V_D \cup V_{term}, E_D \cup E_W \cup E_S)$ where the nodes correspond to sequences and the edges to priority conflicts:

- V_D contains for all $\boldsymbol{x}^j \in \mathcal{X}' \setminus \mathcal{X}'_{term}$ and the difference vector $\boldsymbol{d}^j = \mathrm{succ}_{\mathcal{X}'}(\boldsymbol{x}^i) - \boldsymbol{x}^i$, for all $\boldsymbol{\lambda} \in \Lambda(\boldsymbol{d}^j)$ and all permutations π of $\mathcal{R}(\boldsymbol{d}^j, \boldsymbol{\lambda})$ the sequence $\sigma_{\pi,\boldsymbol{\lambda}}(\boldsymbol{x}^j, \boldsymbol{d}^j)$.
- V_{term} contains for all $\boldsymbol{x}^k \in \mathcal{X}'_{term}$ the trivial sequence $\sigma(\boldsymbol{x}^k, \boldsymbol{0})$.
- E_D contains all edges between two sequences σ, σ' stemming from the same difference vector.
- E_S reflects all strong priority conflicts between sequences σ, σ' stemming from distinct difference vectors.
- E_W reflects all weak priority conflicts between sequences σ, σ' stemming from distinct difference vectors.

The edges in E_D induce a clique partition $\mathcal{Q}$ of $\mathcal{G}$ in as many cliques [3] as there are observed states in $\mathcal{X}' \setminus \mathcal{X}'_{term}$ resp. difference vectors in $\mathcal{D}$: $V_{\mathcal{D}} = Q_1 \cup \ldots \cup Q_{|\mathcal{D}|}$. Moreover, each node in V_{term} corresponds to a clique of size 1, so that $\mathcal{G}$ is partitioned into $|\mathcal{X}'|$ many cliques.

Example 5. The resulting priority conflict graph $\mathcal{G}$ of the running example is shown in Fig. 2.

[3] A clique is a subset of mutually adjacent nodes.

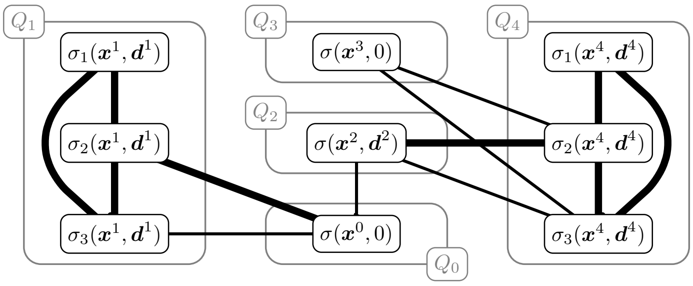

Fig. 2. The conflict graph resulting from the sequences listed in Exp. 4, where bold edges indicate SPCs, thin edges WPCs and gray boxes the clique partition.

Selection of suitable sequences. To obtain a network explaining all observations, we have to select one sequence per difference vector $\boldsymbol{d}^j$, i.e., exactly one node from each clique $Q_j \in \mathcal{Q}$. To encode the priority conflicts involving terminal states, we require also to select all trivial sequences $\sigma(\boldsymbol{x}^k, \boldsymbol{0})$, i.e., all nodes from V_{term}. Thus, we are interested in subsets $S \subseteq V_D$ of cardinality $|\mathcal{D}|$ such that $|S \cap Q_j| = 1$ for each $Q_j \in \mathcal{Q}$, and no SPCs occur in $S \subseteq V_{term}$.
The set of all such solutions $S \cup V_{term}$ can be encoded by all vectors $\boldsymbol{x} \in \{0,1\}^{|V_D \cup V_{term}|}$ satisfying

$$\begin{aligned} \textstyle\sum_{\sigma \in Q_j} \boldsymbol{x}_\sigma &= 1 && \forall Q_j \in \mathcal{Q} \\ \boldsymbol{x}_\sigma &= 1 && \forall \sigma \in V_{term} \\ \boldsymbol{x}_\sigma + \boldsymbol{x}_{\sigma'} &\leq 1 && \forall \sigma\sigma' \in E_S \\ \boldsymbol{x}_\sigma &\in \{0,1\} && \forall \sigma \in V_D \cup V_{term}. \end{aligned}$$

Example 6. From $\mathcal{G}$ in Exp. 5, we obtain the following feasible subsets $S_i \cup V_{term}$

$S_1 = \{\sigma_1(\boldsymbol{x}^1, \boldsymbol{d}^1), \sigma(\boldsymbol{x}^2, \boldsymbol{d}^2), \sigma_1(\boldsymbol{x}^4, \boldsymbol{d}^4)\}$, $S_3 = \{\sigma_1(\boldsymbol{x}^1, \boldsymbol{d}^1), \sigma(\boldsymbol{x}^2, \boldsymbol{d}^2), \sigma_3(\boldsymbol{x}^4, \boldsymbol{d}^4)\}$,
$S_2 = \{\sigma_3(\boldsymbol{x}^1, \boldsymbol{d}^1), \sigma(\boldsymbol{x}^2, \boldsymbol{d}^2), \sigma_1(\boldsymbol{x}^4, \boldsymbol{d}^4)\}$, $S_4 = \{\sigma_3(\boldsymbol{x}^1, \boldsymbol{d}^1), \sigma(\boldsymbol{x}^2, \boldsymbol{d}^2), \sigma_3(\boldsymbol{x}^4, \boldsymbol{d}^4)\}$.

Composition of conformal networks. Every selected subset $S \cup V_{term}$ corresponds to a standard network $\mathcal{P}_S = (P, T_S, A_S, w)$ which is conformal with $\mathcal{X}'$ (but not yet necessarily $\mathcal{X}'$-deterministic):

- we obtain the columns of the incidence matrix M_S of the network by taking the union of all sets $\mathcal{R}(\boldsymbol{d}^j, \boldsymbol{\lambda})$ corresponding to the sequences $\sigma = \sigma_{\pi,\boldsymbol{\lambda}}(\boldsymbol{x}^j, \boldsymbol{d}^j)$ selected by $\sigma \in S$;
- there might be weak priority conflicts $\sigma\sigma' \in E_W$ for nodes $\sigma, \sigma' \in S \cup V_{term}$ which have to be resolved subsequently by inserting appropriate control-arcs.

Example 7. We apply the method only to the feasible set $S_1 \cup V_{term}$ from Exp. 6 (all solutions for $S_2 \cup V_{term}$, $S_3 \cup V_{term}$ and $S_4 \cup V_{term}$ are presented in Exp. 9). We construct the standard network presented in Fig. 3 with $T_{S_1} = \{\boldsymbol{d}^1, \boldsymbol{d}^2, \boldsymbol{d}^4\}$. There is a priority conflict between $\sigma(\boldsymbol{x}^2, \boldsymbol{d}^2)$ and $\sigma(\boldsymbol{x}^0, 0)$ due to $\boldsymbol{d}^2, 0 \in T(\boldsymbol{x}^2) \cap T(\boldsymbol{x}^0)$.

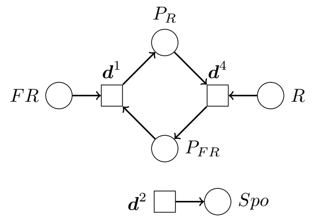

Fig. 3. Standard network $\mathcal{P}_{S_1} = (P, T_{S_1}, A_{S_1}, w)$ from solution S_1 (Exp. 6)

3.5 Inserting control-arcs.

For each of the yet reconstructed standard networks $\mathcal{P}_S = (P, T_S, A_S, w)$ resulting from a selected subset S from the previous reconstruction step, we have to determine appropriate control-arcs in order to resolve weak priority conflicts corresponding to edges $\sigma\sigma' \in E_W$ with $\sigma, \sigma' \in S \cup V_{term}$ (if any), in order to turn $\mathcal{P}_S$ into a catalytic conformal extended Petri net $\mathcal{P}_S = (P, T_S, \mathcal{A}_S, w)$.

Recall that we have a weak priority conflict between two sequences σ and σ' if there are update vectors $\boldsymbol{r}^t \neq \boldsymbol{r}^{t'}$ and intermediate states $\boldsymbol{y} \neq \boldsymbol{y}'$ with $t, t' \in T(\boldsymbol{y}) \cap T(\boldsymbol{y}')$ such that $(\boldsymbol{y}, \boldsymbol{r}^t) \in \sigma$ but $(\boldsymbol{y}', \boldsymbol{r}^{t'}) \in \sigma'$. This weak priority conflict has to be resolved by adding appropriate control-arcs such that

- the update vector $\boldsymbol{r}^t$ becomes a transition t with $t \in T_{\mathcal{A}}(\boldsymbol{y})$ but $t \notin T_{\mathcal{A}}(\boldsymbol{y}')$ (or vice versa) if $\boldsymbol{y}, \boldsymbol{y}' \notin \mathcal{X}'_{term}$ or
- the update vector $\boldsymbol{r}^t$ becomes a transition t with $t \in T_{\mathcal{A}}(\boldsymbol{y})$ which is disabled by control-arcs in $\boldsymbol{y}'$ if $\boldsymbol{y}' \in \mathcal{X}'_{term}$.

Inserting control-arcs This task can be done by using similar techniques as proposed in [1,2]. Let $P(\boldsymbol{y}, \boldsymbol{y}')$ be the set of places where $\boldsymbol{y}$ and $\boldsymbol{y}'$ differ, i.e., $P(\boldsymbol{y}, \boldsymbol{y}') = \{p \in P : y_p \neq y'_p\}$. In order to disable transition t resulting from $\boldsymbol{r}^t$ at $\boldsymbol{y}'$, we can include either

- a read-arc $(p, t) \in A_R$ with weight $w(p, t) > y'_p$ for some $p \in P(\boldsymbol{y}, \boldsymbol{y}')$ with $y_p > y'_p$ or
- an inhibitor-arc $(p, t) \in A_I$ with weight $w(p, t) < y_p$ for some $p \in P(\boldsymbol{y}, \boldsymbol{y}')$ with $y_p < y'_p$.

Each of the so-determined control-arcs defines a transition t with the desired properties (inheriting the standard arcs from $\boldsymbol{r}^t$ and having either a read-arc or an inhibitor-arc as described above).

Remark 2. In case of a SPC involving states $\boldsymbol{y} = \boldsymbol{y}'$, the set $P(\boldsymbol{y}, \boldsymbol{y}')$ becomes empty and it is, therefore, not possible to resolve a SPC by control-arcs.

For every reconstructed standard network $\mathcal{P}_S = (P, T_S, A_S, w)$ and any subset $P' \subseteq P$ containing exactly one place from $P(\boldsymbol{y}, \boldsymbol{y}')$ for every weak priority conflict, we get a catalytic conformal extended Petri net $\mathcal{P}_{S,P'} = (P, T_S, \mathcal{A}_{S,P'}, w)$ by inserting the respective control-arcs for all $p \in P'$.

Example 8. We define control-arcs to resolve the WPC between $\sigma(\boldsymbol{x}^2, \boldsymbol{d}^2)$ and $\sigma(\boldsymbol{x}^0, 0)$ for the network $\mathcal{P}_{S_1}$. We obtain

$$P(\boldsymbol{x}^2, \boldsymbol{x}^0) = \{P_{FR}, P_R\} \text{ by } \boldsymbol{x}^2 = (0, 0, \mathbf{0}, \mathbf{1}, 0)^T \text{ and } \boldsymbol{x}^0 = (0, 0, \mathbf{1}, \mathbf{0}, 0)^T.$$

Any non-empty subset of $P(\boldsymbol{x}^2, \boldsymbol{x}^0)$ can be used to disable $\boldsymbol{d}^2$ at $\boldsymbol{x}^0 \in \mathcal{X}'_{term}$. For P_{FR}, $\boldsymbol{x}^2_{P_{FR}} < \boldsymbol{x}^0_{P_{FR}}$ holds, leading to an inhibitor-arc $(P_{FR}, \boldsymbol{d}^2) \in \mathcal{A}_{S_1,P'}$, and for P_R, $\boldsymbol{x}^2_{P_R} > \boldsymbol{x}^0_{P_R}$ holds, leading to a read-arc $(P_R, \boldsymbol{d}^2) \in \mathcal{A}_{S_1,P'}$ both with weight 1. The two possible alternatives are presented in Fig. 4.

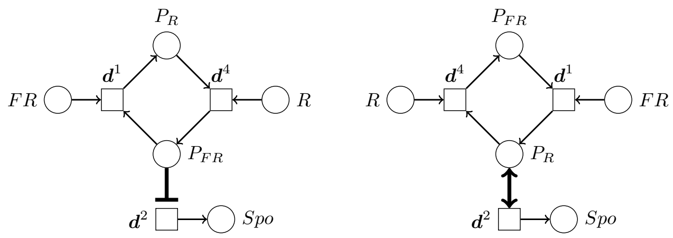

Fig. 4. The two catalytic conformal networks resulting from $\mathcal{P}_{S_1}$ in Exp. 8

3.6 Determining priority relations

To generate the required priorities for each of the yet reconstructed extended networks $\mathcal{P}_{S,P'} = (P, T_S, \mathcal{A}_{S,P'}, w)$, we only need to set the priorities among all the transitions in T_S according to the sequences selected for S.

Recall that every $\sigma \in S$ stands for a sequence

$$\sigma = \sigma_{\pi,\boldsymbol{\lambda}}(\boldsymbol{x}^j, \boldsymbol{d}^j) = \big((\boldsymbol{y}^1, \boldsymbol{r}^{t_1}), (\boldsymbol{y}^2, \boldsymbol{r}^{t_2}), \ldots, (\boldsymbol{y}^m, \boldsymbol{r}^{t_m})\big)$$

which induces a priority relation $\mathcal{O}_\sigma$ indicating that the transition t_i resulting from $\boldsymbol{r}^{t_i}$ is supposed to have highest priority at $\boldsymbol{y}^i$. That is, $\mathcal{O}_\sigma$ is defined by

$$\mathcal{O}_\sigma = \Big\{t_i > t : t \in T_{\mathcal{A}_{S,P'}}(\boldsymbol{y}^i) \setminus t_i, 1 \le i \le m\Big\}.$$

By construction, there are no priority conflicts in the extended network $\mathcal{P}_{S,P'}$ between $\mathcal{O}_\sigma$ and $\mathcal{O}_{\sigma'}$ for any $\sigma, \sigma' \in S$, thus we obtain the studied partial order

$$\mathcal{O}_{S,P'} = \bigcup_{\sigma \in S} \mathcal{O}_\sigma.$$

This implies finally that every extended network $\mathcal{P}_{S,P'} = (P, T_S, \mathcal{A}_{S,P'}, w)$ together with the partial order $\mathcal{O}_{S,P'}$ constitutes an $\mathcal{X}'$-deterministic extended Petri net fitting the given data $\mathcal{X}'$.

Example 9. For the running example, it is left to determine the priority relations. For the extended Petri nets $\mathcal{P}_{S_1,P'}$, we can easily verify that $T_{\mathcal{A}_{S_1,P'}}(\boldsymbol{x})$ contains exactly one transition for all $\boldsymbol{x} \in \mathcal{X}'$, so no priorities are implied and $\mathcal{O}_{S_1,P'} = \emptyset$ follows. For the Petri nets coming from the other sets S_2, S_3, S_4, all possible minimal $\mathcal{X}'$-deterministic extended Petri nets are depicted in Fig. 5, 6 and 7.

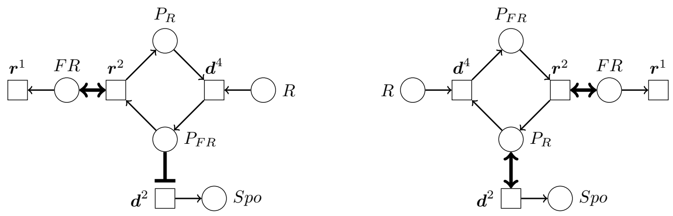

Fig. 5. From S_2, two catalytic conformal networks $\mathcal{P}_{S_2}$ result, both with priority relations $\mathcal{O}_2 = \{(\boldsymbol{r}^2 > \boldsymbol{r}^1)\}$.

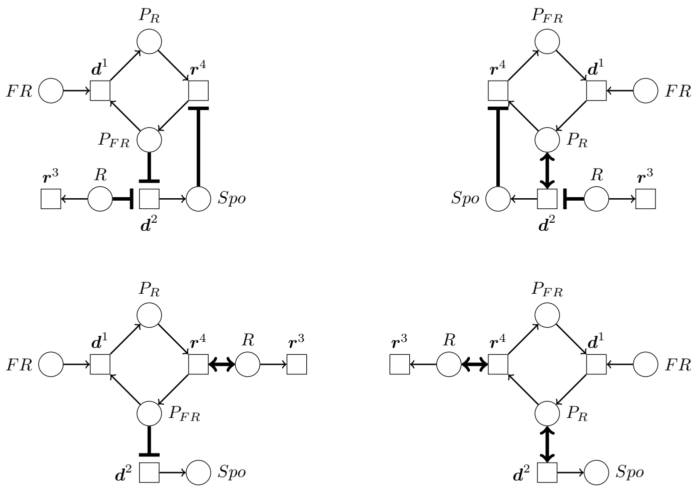

Fig. 6. From S_3, four minimal catalytic conformal networks $\mathcal{P}_{S_3}$ result, all with priority relations $\mathcal{O}_3 = \{(\boldsymbol{r}^4 > \boldsymbol{r}^3)\}$.

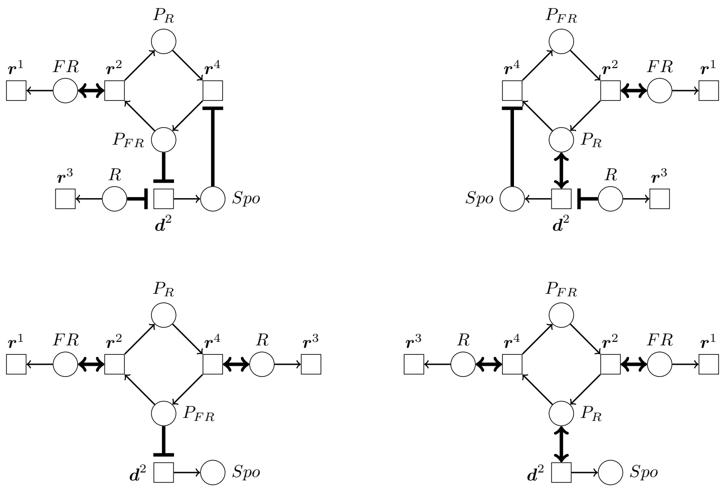

Fig. 7. From S_4, four minimal catalytic conformal networks $\mathcal{P}_{S_4}$ result, all with priority relations $\mathcal{O}_4 = \{(\boldsymbol{r}^2 > \boldsymbol{r}^1), (\boldsymbol{r}^4 > \boldsymbol{r}^3)\}$.

4 Concluding remarks

To summarize, we present in this paper the steps of an integrative reconstruction method to generate all possible $\mathcal{X}'$-deterministic extended Petri nets from monotone and reproducible experimental time-series data $\mathcal{X}'$.

This approach is based on previous works for special cases: the reconstruction of standard networks [4], standard networks with priorities [8] and extended Petri nets [1,2]. Here, we modify and generalize the previous methods by

- adjusting the representation of the observed difference vectors $\boldsymbol{d}^j$ to the case of extended networks with priorities (where $\boldsymbol{d}^j$ might be enabled at a terminal state in $\mathcal{X}'_{term}$),
- refining the idea from [8] to construct a priority conflict graph by distinguishing weak and strong priority conflicts (where only strong conflicts affect the selection),
- generalizing the method from [1,2] such that weak priority conflicts can be resolved by inserting control-arcs (where arbitrary arcs weights can occur).

Note that a preprocessing (to test the experimental data $\mathcal{X}'$ for reproducibility and, if necessary, to handle infeasible situations) can be handled similar as in [4] and a postprocessing (to keep only "minimal" solutions in the sense that all other $\mathcal{X}'$-deterministic extended Petri nets fitting the data contain the returned ones) is presented in [13].

In total, this integrative approach is promising for the reconstruction of networks fully fitting the experimentally observed phenomena.

Our further goal is to make the new approach accessible by a suitable implementation, e.g., using Answer Set Programming as in the case of the reconstruction of standard networks with priorities [7] and to apply it to new biological experimental data.

References

1. M. Durzinsky, W. Marwan, and A. K. Wagler. Reconstruction of extended Petri nets from time series data and its application to signal transduction and to gene regulatory networks. *BMC Systems Biology*, 5, 2011.
2. M. Durzinsky, W. Marwan, and A. K. Wagler. Reconstruction of extended Petri nets from time-series data by using logical control functions. *Journal of Mathematical Biology*, 66:203–223, 2013.
3. M. Durzinsky, A. K. Wagler, and R. Weismantel. A combinatorial approach to reconstruct Petri nets from experimental data. In Monika Heiner and Adelinde M. Uhrmacher, editors, *CMSB*, volume 5307 of *Lecture Notes in Computer Science*, pages 328–346. Springer, 2008.
4. M. Durzinsky, A. K. Wagler, and R. Weismantel. An algorithmic framework for network reconstruction. *Journal of Theoretical Computer Science*, 412(26):2800–2815, 2011.
5. I. Koch and M. Heiner. Petri nets. In B. H. Junker, F. Schreiber, editors, *Biological Network Analysis*, Wiley Book Series on Bioinformatics, pages 139–179, 2007.
6. T. Lamparter and W. Marwan. Spectroscopic detection of a phytochrome-like photoreceptor in the myxomycete physarum polycephalum and the kinetic mechanism for the photocontrol of sporulation. *Photochem Photobiol*, 73:697–702, 2001.
7. M. Durzinsky M. Ostrowski, T. Schaub and W. Marwan. Automatic network reconstruction using ASP. *Th. and Practice of Logic Progr.*, 11:749–766, 2011.
8. W. Marwan, A. K. Wagler, and R. Weismantel. A mathematical approach to solve the network reconstruction problem. *Math. Methods of Operations Research*, 67(1):117–132, 2008.
9. J. W. Pinney, R. D. Westhead, and G. A. McConkey. Petri net representations in systems biology. *Biochem. Soc. Tarns.*, 31:1513–1515, 2003.
10. C. Starostzik and W. Marwan. Functional mapping of the branched signal transduction pathway that controls sporulation in *Physarum polycephalum. Photochem. Photobiol.*, 62:930–933, 1995.
11. L. M. Torres and A. K. Wagler. Encoding the dynamics of deterministic systems. *Math. Methods of Operations Research*, 73:281–300, 2011.
12. A. K. Wagler. *Prediction of network structure*, In: Modeling and System Biology, I. Koch, F. Schreiber, W. Reisig (eds.) *Computational Biology 16*. Springer London, pages 309–338 2010.
13. A. K. Wagler and J.-T. Wegener. On minimality and equivalence of Petri nets. *Proceedings of Concurrency, Specification and Programming CS&P'2012 Workshop*, 2:382–393, 2012.