

Reconfigurable Petri Nets with Transition Priorities and Inhibitor Arcs

Julia Padberg^(✉)

Hamburg University of Applied Sciences, Hamburg, Germany
julia.padberg@haw-hamburg.de

Abstract. In this paper we introduce additional control structures for reconfigurable Petri nets. The main contributions are inhibitor arcs and transition priorities for reconfigurable Petri nets. The first ensure that a marking can inhibit the firing of a transition. Inhibitor arcs allow a transition to fire only if the adjacent place is empty. Transition priorities are given by an order of transitions and restrict the firing as well. A transition may fire only if it has the highest priority of all enabled transitions. Both concepts are compatible with reconfigurable Petri nets. In this paper we prove that place/transitions nets with inhibitor arcs and with transition priorities yield \mathcal{M} -adhesive categories. Hence, we obtain the well-known results for \mathcal{M} -adhesive categories. Moreover, we state the extension of our results to other types of Petri nets.

We illustrate the new concepts within an ongoing case study concerning travel agencies. This study deals with the organisation of processes that are constantly suspended by others. The main focus of the case study is to investigate the possibilities of small and medium travel agencies to provide a continuous service for their customers while travelling.

Keywords: Reconfigurable Petri nets · Category of partially ordered sets · Inhibitor arcs · Transition priorities · \mathcal{M} -adhesive transformation system

1 Introduction

Reconfigurable Petri nets consist of a Petri net with a marking and a set of rules whose application modifies the net's structure at runtime. Typical application areas are concerned with the modelling of dynamic structures, for example workflows in a dynamic infrastructure. They can be considered to be a family of formal modelling techniques based on different types of Petri nets (for example in [11, 12, 16, 20, 26]). Their motivation is the observation that in increasingly many application areas the underlying system has to be dynamic in a structural sense. Complex coordination and structural adaptation at run-time (e.g. mobile ad-hoc networks, communication spaces, ubiquitous computing) are main features that need to be modelled adequately. The distinction between the net behaviour and the dynamic change of its net structure is the characteristic feature that makes reconfigurable Petri nets so suitable for systems with dynamic structures.

In a motivating example (see Sect. 3) the two possibilities of modelling change are used for separating common processes from additional processes being introduced for special purposes. The main problem is the best order of the assignment of pending tasks to the employees. In this case study we found that the use of transition priorities is really helpful. Reconfigurable Petri nets are given as a transformation system, that is formulated in terms of category theory, so called \mathcal{M} -adhesive transformation systems. Transition priorities are easy to define but they are difficult to integrate into \mathcal{M} -adhesive categories. In Sect. 5 we investigate the category of partial orders, prove it to be an \mathcal{M} -adhesive category and obtain by comma category constructions the intended category of PT nets with transition priorities. Inhibitor arcs can be achieved by a straightforward extension of Petri nets, so they are presented in brief. Since we use algebraic high-level nets in the case study, we discuss the transfer of results for PT nets to other kinds of Petri nets.

The paper is organized as follows: First we summarize reconfigurable place/transition (PT) nets (see Sect. 2). Then an example is presented in Sect. 3 motivating the addition of transition priorities. A short review of \mathcal{M} -adhesive transformation systems is given in Sect. 4. Subsequently, Sect. 5 extends the transitions with a partial order, describing the priorities between the transitions. We employ the category of partial orders **PoSets** and obtain an \mathcal{M} -adhesive category of PT nets with transition priorities. Then (in Sect. 6) we add inhibitor arcs to PT nets and show that they yield an \mathcal{M} -adhesive category as well. Related work concerns other kinds of control structures for reconfigurable Petri nets (see Sect. 7) and we close with some remarks on future work.

2 Reconfigurable Petri Nets

In this section we introduce reconfigurable Petri nets, where the main focus is on place/transition nets, but other types of Petri nets are discussed as well.

2.1 Reconfigurable Place/Transition Nets

In the algebraic approach to Petri nets a (marked) place/transition net is given by $N = (P, T, pre, post, M_0)$ with pre- and post-domain functions $pre, post : T \rightarrow P^\oplus$ and an initial marking $M_0 \in P^\oplus$, where P^\oplus is the free commutative monoid over the set P of places. For $M_1, M_2 \in P^\oplus$ we have $M_1 \leq M_2$ if $M_1(p) \leq M_2(p)$ for all $p \in P$. A transition $t \in T$ is M -enabled for a marking $M \in P^\oplus$ if we have $pre(t) \leq M$, and in this case the follower marking M' is given by $M' = M \ominus pre(t) \oplus post(t)$ and $M[t]M'$ is called firing step.

Net morphisms map places to places and transitions to transitions. They are given as a pair of mappings for the places and the transitions, so that the net structure is preserved. Given two (PT) place/transition nets $N_i = (P_i, T_i, pre_i, post_i, M_i)$ for $i \in \{1, 2\}$ a net morphism $f : N_1 \rightarrow N_2$ is defined by $f = (f_P : P_1 \rightarrow P_2, f_T : T_1 \rightarrow T_2)$, so that the following equations hold:

1. $pre_2 \circ f_T = f_P^\oplus \circ pre_1$ and $post_2 \circ f_T = f_P^\oplus \circ post_1$
2. $M_1(p) \leq M_2(f_P(p))$ for all $p \in P_1$

Moreover, the morphism f is called strict if both f_P and f_T are injective and $M_1(p) = M_2(f_P(p))$ holds for all $p \in P_1$. PT nets together with net morphisms comprise the category **PT**.

\mathcal{M} -adhesive transformation systems (see Sect. 4) can be considered as a unifying framework for graph and Petri net transformations providing enough structure that most notions and results from algebraic graph transformation systems are available, as results on parallelism and concurrency of rules and transformations, results on negative application conditions and constraints, and many more results (see [7, 8]). A rule in the DPO approach is given by three nets called left hand side L , interface K and right hand side R , respectively, and a span of two strict net morphisms $K \rightarrow L$ and $K \rightarrow R$.

Additionally, an occurrence morphism $o : L \rightarrow N$ is required that identifies the relevant parts of the left hand side in the given net N . Then a transformation step $N \xrightarrow{(r,o)} M$ via rule r can be constructed in two steps. Given a rule with an occurrence $o : L \rightarrow N$ the gluing conditions have to be satisfied in order to apply a rule at a given occurrence. These conditions ensure the result is again a well-defined net. It is a sufficient condition for the existence and uniqueness of the so-called pushout complement which is needed for the first step in a transformation.

In this case, we obtain a net M leading to a direct transformation $N \xrightarrow{(r,o)} M$ consisting of the following pushouts (1) and (2) in Fig. 1. This construction as well as a huge amount of notion and results are available since PT nets can be proven to be an \mathcal{M} -adhesive transformation category (see [7]). Hence we can combine one net together with a set of rules leading to reconfigurable PT nets.

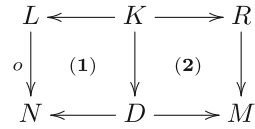


Fig. 1. Transformation of a net

Definition 1 (Reconfigurable PT Nets). A reconfigurable PT net $RN = (N, \mathcal{R})$ is given by a PT net N and a set of rules \mathcal{R} .

2.2 Other Types of Reconfigurable Petri Nets

Decorated place/transition nets [22] extend PT nets. They are the basis for the tool RECONNET[6] and provide additional decorations for PT nets: capacities, names for places as well as transitions and additional transition labels that can be changed by firing that transition. This last concept allows a better coordination of transition firing and rule application, for example to ensure that a transition has fired (repeatedly) before a transformation step may take place. This extension is conservative with respect to PT nets as it does not change the net behaviour, but it is crucial for the application of the rules and provides the possibility to control the application of rules. Decorated PT nets together with net morphisms, which additionally preserve the decorations, yield the category **decoPT**.

Algebraic high-level (AHL) nets (as used in Sect. 3) extend PT nets by allowing net inscriptions with terms over a given signature Σ and by interpreting tokens as data elements over a Σ -algebra. An AHL net $AN = (\Sigma, P, T, pre, post, cond, type, A, M_0)$ consists of

- an algebraic signature $\Sigma = (S, OP, X)$ with additional variables X ,
- a set of places P and a set of transitions T ;
- pre- and post domain functions $pre, post : T \rightarrow (T_\Sigma(X) \otimes P)^\oplus$;
- firing conditions $cond : T \rightarrow \mathcal{P}_{fin}(Eqns(\Sigma; X))$;
- a type of places $type : P \rightarrow S$;
- a (Σ, E) -algebra A and
- a marking $M_0 \in (A \otimes P)^\oplus$,

where $T_\Sigma(X)$ is the set of terms with variables over X , $(T_\Sigma(X) \otimes P) = \{(term, p) | term \in T_\Sigma(X)_{type(p)}, p \in P\}$, $(A \otimes P) = \{(a, p) | a \in A_{type(p)}, p \in P\}$ and $Eqns(\Sigma; X)$ are all equations over the signature Σ with variables X . An AHL-net morphism $f : AN_1 \rightarrow AN_2$ is given by $f = (f_P, f_T)$ with functions $f_P : P_1 \rightarrow P_2$ and $f_T : T_1 \rightarrow T_2$ satisfying

- (1) $(id \otimes f_P)^\oplus \circ pre_1 = pre_2 \circ f_T$ and $(id \otimes f_P)^\oplus \circ post_1 = post_2 \circ f_T$,
- (2) $cond_2 \circ f_T = cond_1$ and
- (3) $type_2 \circ f_P = type_1$.

and leads to the category defined by AHL nets and AHL net morphisms, denoted by **AHL**.

3 Motivating Example

In this section we give a motivating example taken from a case study concerning workflows in a small travel agency.

The increasing possibilities of the internet to plan journeys and to buy travel product has led to a severe decline of the market for travel agencies. In Germany there are about 9.729 travel agencies in 2013, down from 13.753 in 2004 (see [29]). Today one of the buzz words in the travel and tourism industry is personalized traveller journey. The main focus of the major computer reservations systems companies, also known as global distribution systems (e.g., Galileo, Amadeus, SABRE) shifts to delivering continually and consistently the products and services directly to customers. Due to technological change and the demand for more personalization many global distribution systems increasingly sell products to travellers directly (see e.g. [31]). This tendency is a further threat for small and medium travel agencies. Our case study aims at supporting small and medium travel agencies to provide direct and instantaneous personal support of their customers. So, they can compete against larger companies by providing truly personal support. This results in complex and suspended processes for the employees of the travel agency, the travel agents.

We investigate the actual processes and model them as algebraic high-level nets. In Fig. 2 a reduced example is given. The additional tasks that arise through the new task of personal support of travelling customers are modelled by rules (see Fig. 3) that suspend the actual process and implement the new tasks. In Fig. 2 an algebraic high-level (AHL) net is given, that illustrates some of the tasks of the travel agents. The signature and the algebra are merely hinted at and deliver the inscriptions and the data elements used in the net. The net inscriptions comprise sorts that determine the type of the tokens in a place, arc inscriptions with terms and firing conditions given as equations in the left bottom of the transition box. The tokens are data elements of the algebra. A travel agent's role is to help customers plan, choose and arrange their travel. In our case study the main focus is the advising of customers. This can be done via phone, email or directly when the customer is in the agency's office. In principle the travel agent is either in direct contact with the customer or is concerned with

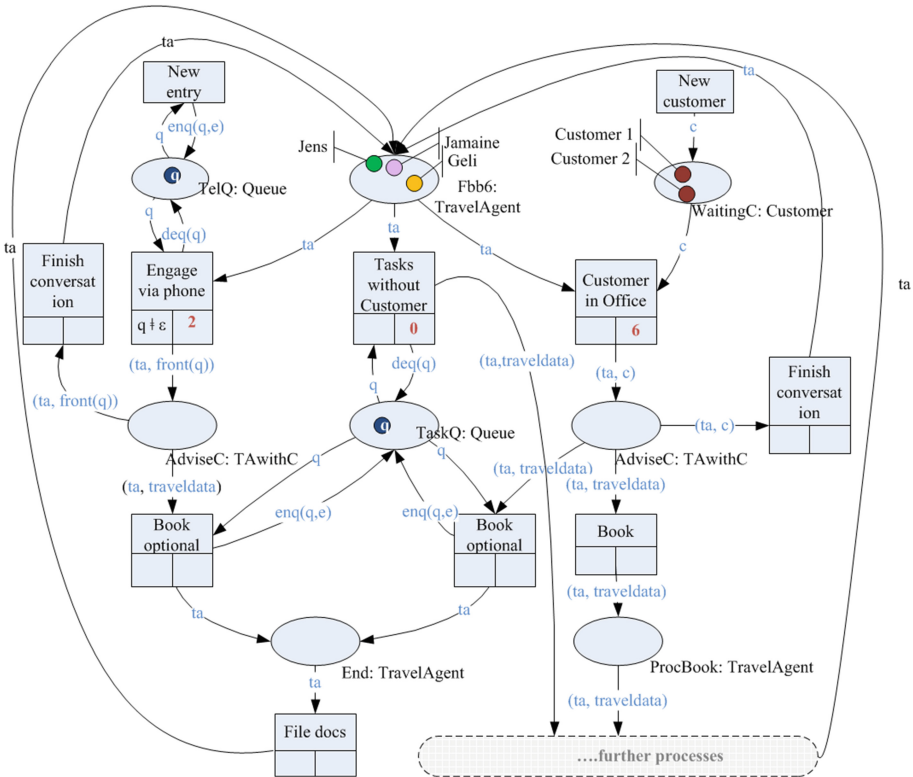


Fig. 2. Simplified model of processes in the travel agency

tasks as booking options, confirmation by the customer, travel confirmation, ticket issuing, ticket delivery to customers, filing, updating customer payments, accounting organizers payment, and so on. If the agent is idle there are clear priorities, the customer in the agency is served first, then the phone queue and subsequently the email requests (not illustrated in Fig. 2) are processed, and at last the tasks without customer contact are taken up. This order of tasks is modelled by the transitions (the boxes in Fig. 2) and their priorities (the number in the right bottom of the transition box). Since not all transitions are provided with priorities, those without are not considered for the prioritisation.

The signature of an AHL net provides the syntax for the net inscriptions, i.e., sorts, terms and equations. The signature **TA** is sketched here by **TA = Queue + Customer + TravelData +**
sorts : TravelAgent, TAwithC, ...
ops : (–, –) : TravelAgent × Customer → TAwithC

The **TA**-algebra $A = (A_{TravelAgent}, A_{Customer}, \dots)$ consists of sets and operations according to the signature:

- $A_{TravelAgent} = \{Jens, Geli, Jamaine, Torsten, \dots\}$,
- $A_{Customer}$ the set of all customers and so on...

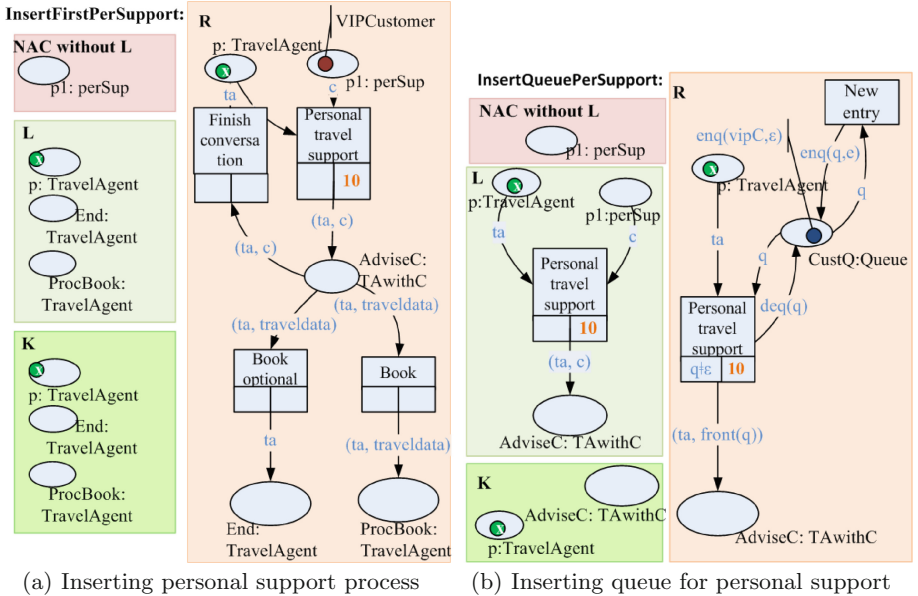


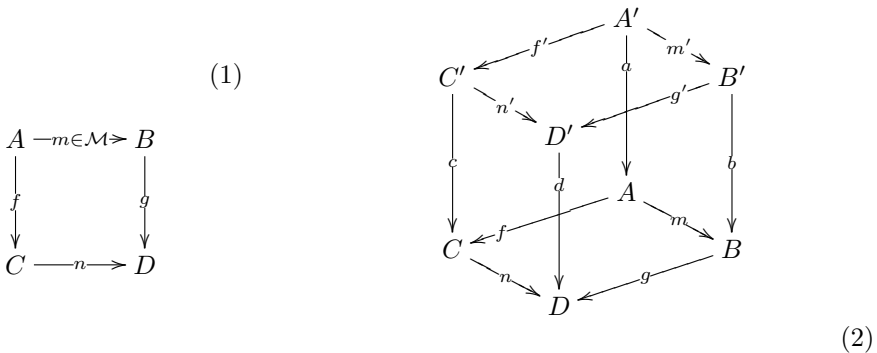
Fig. 3. Exemplary rules

This AHL net models essentially the processes that occur without the additional tasks of supporting travelling customers. The following rules model the insertion of processes that deal with handling the interruption by a VIP-customer that is dealt on demand. Nevertheless, the process of advising customers in person or by phone, cannot be suspended. Hence, the rules can only be applied to places that are not typed with the sort $TAwithC$. In Fig. 3 two exemplary rules are given, where parts of the negative application conditions NAC are omitted. The rule in Fig. 3(a) models the insertion of the personal support process for the first customer. The rule in Fig. 3(b) models the insertion of a queue for processing the personal support. In both cases a transition is inserted into the AHL with a higher priority than the other transitions in the net.

4 Review of \mathcal{M} -Adhesive Transformation Systems

The theory of \mathcal{M} -adhesive transformation systems¹ has been developed as an abstract framework for different types of graph and Petri net transformation systems [7, 10]. They have been instantiated with various graphs, e.g., hypergraphs, attributed and typed graphs, but also with structures, algebraic specifications and various Petri net classes, as elementary nets, place/transition nets, Colored Petri nets, or algebraic high-level nets [7]. The fundamental construct for \mathcal{M} -adhesive categories and systems are \mathcal{M} -van Kampen squares [10, 18]²

Definition 2 (\mathcal{M} -Van Kampen square). A pushout (1) with $m \in \mathcal{M}$ is an \mathcal{M} -van Kampen square, if for any commutative cube (2) with (1) in the bottom the back faces being pullbacks, the following holds: the top is pushout \Leftrightarrow the front faces are pullbacks.



\mathcal{M} -adhesive transformation systems can be seen as an abstract transformation systems in the double pushout approach based on \mathcal{M} -adhesive categories [10].

Definition 3 (\mathcal{M} -Adhesive Category). A class \mathcal{M} of monomorphisms in \mathbf{C} is called *PO-PB compatible*, if

¹ See page 2 in [9] for the relation to other types of HLR systems.

² For a discussion of the various adhesive categories see page 6 in [9].

1. Pushouts along \mathcal{M} -morphisms exist and \mathcal{M} is stable under pushouts.
2. Pullbacks along \mathcal{M} -morphisms exist and \mathcal{M} is stable under pullbacks.
3. \mathcal{M} contains all identities and is closed under composition.

Given a PO-PB compatible class \mathcal{M} of monomorphisms in \mathbf{C} , then $(\mathbf{C}, \mathcal{M})$ is called \mathcal{M} -adhesive category, if pushouts along \mathcal{M} -morphisms are \mathcal{M} -Van Kampen squares (see Definition 2). An \mathcal{M} -adhesive transformation system $AHS = (\mathbf{C}, \mathcal{M}, P)$ consists of an \mathcal{M} -adhesive category $(\mathbf{C}, \mathcal{M})$ and a set of rules P .

Remark 1. The following kinds of Petri nets yield \mathcal{M} -adhesive categories:

- PT nets and morphisms as given in Sect. 2 yield an \mathcal{M} -adhesive category **PT** (see [7]).
- Algebraic high-level nets as given in Sect. 2 have been shown in [25] to be an \mathcal{M} -adhesive category **AHL** for \mathcal{M} being the class of strict morphisms³.
- In [22] it is shown that decorated place/transition nets yield an \mathcal{M} -adhesive transformation category **decoPT** for \mathcal{M} being the corresponding class of strict morphisms.

5 Transition Priorities

The set of transitions T is equipped with a partial order \leq on the transitions. A transition t is enabled under a marking M , if $pre(t) \geq M$ and if there is no t' being enabled under M so that $t \leq t'$. As we have discussed in Sect. 3 this allows using priorities only for a subset of transitions and extends the original approach [2] to transition priorities.

We first need to investigate the category **PoSets** of partially ordered sets. Note, it is not a partial order considered to be a category, but the category of all partial orders with order-preserving maps as morphisms. In [5] this category has been examined thoroughly.

Definition 4 (Category PoSets). *The objects are partially ordered sets, given by a set P and a partial order \leq over P . The morphisms in this category are order-preserving maps, that are maps $f : P_1 \rightarrow P_2$ preserving the order, so $x \leq y$ implies $f(x) \leq f(y)$.*

Composition and identity are defined as for sets and are both order-preserving, so **PoSets** is indeed a category [5]. The relation to the category of sets can be given by an adjunction.

Lemma 1 (Adjunction between Sets and PoSets). *The left adjoint functor $F : \mathbf{Sets} \rightarrow \mathbf{PoSets}$ is given by $F(S \xrightarrow{f} S') = (S, ID_S) \xrightarrow{f} (S', ID_{S'})$ where ID_S is the identity relation over a set S . The right adjoint functor $G : \mathbf{PoSets} \rightarrow \mathbf{Sets}$ is defined by $G((P, \leq_P) \xrightarrow{g} (P', \leq_{P'})) = P \xrightarrow{g} P'$.*

³ In [25] they are called AHL-systems with morphisms that are isomorphisms on the algebra part.

The counit is the natural transformation $\epsilon : F \circ G \rightarrow \mathbf{1}_{\mathbf{PoSets}}$ with $\epsilon_M = id_M$ an order-preserving map. The unit is the natural transformation $\eta : \mathbf{1}_{\mathbf{Sets}} \rightarrow G \circ F$ with $\eta_M = id_M$.

Proof. Obviously, since the composition of identities leads to the identity with $\epsilon_{F(S)} \circ F(\eta_S) = id_{F(S)} \circ F(id_S) = id_{F(S)}$ and $G(\epsilon_{(S,R)}) \circ \eta_{G(S,R)} = id_{G(S,R)} \circ id_{G(S,R)} = id_{G(S,R)}$.

So, we know that F preserves colimits and G preserves limits. As pushouts are the most prominent construction in the DPO approach, we prove finite cocompleteness by existence of initial objects and pushouts.

Lemma 2 (Initial Object and Pushouts in PoSets).

1. The initial object is (\emptyset, \emptyset) .
2. Given the span $(P_1, \leq_1) \xleftarrow{f} (P_0, \leq_0) \xrightarrow{g} (P_2, \leq_2)$, then there exists the pushout $(P_1, \leq_1) \xrightarrow{g'} (P_3, \leq_3) \xleftarrow{f'} (P_2, \leq_2)$.

Proof. 1. The initial object is (\emptyset, \emptyset) as there is the empty mapping to each partially ordered set in **PoSets** and it is order preserving.

2. For $(P_1, \leq_1) \xleftarrow{f} (P_0, \leq_0) \xrightarrow{g} (P_2, \leq_2)$ there is in **Sets** the span $P_1 \xleftarrow{f} P_0 \xrightarrow{g} P_2$ and its pushout $P_1 \xrightarrow{\bar{g}} \bar{P}_3 \xleftarrow{\bar{f}} P_2$, see pushout (PO) in Diagram 4 and the relation $R_3 \subseteq \bar{P}_3 \times \bar{P}_3$ with

$$\begin{aligned}
 (x_3, y_3) \in R_3 \text{ if and only if} \\
 \exists x_1, y_1 \in P_1 : \bar{g}(x_1) = x_3 \wedge \bar{g}(y_1) = y_3 \wedge x_1 \leq_1 y_1 \quad (3) \\
 \vee \exists x_2, y_2 \in P_2 : \bar{f}(x_2) = x_3 \wedge \bar{f}(y_2) = y_3 \wedge x_2 \leq_2 y_2
 \end{aligned}$$

Since R_3 is not a partial order⁴, we define the relation \bar{R}_3 to be the equivalence closure of all symmetric pairs $\{(x_3, y_3) \mid (x_3, y_3), (y_3, x_3) \in R_3\} \subseteq R_3$. Then we have the quotient $P_3 = \bar{P}_3 / \bar{R}_3$ with $g' := [-] \circ \bar{g} : P_1 \rightarrow P_3$ and $f' := [-] \circ \bar{f} : P_2 \rightarrow P_3$, where $[-] : \bar{P}_3 \rightarrow \bar{P}_3 / \bar{R}_3 = P_3$ is the natural function mapping each element of \bar{P}_3 to its equivalence class.

The relation \leq_3 is the transitive closure of

$$\{(x_3, y_3) \mid x_1 \leq_1 y_1 \text{ for } g'(x_1) = x_3 \text{ and } g'(y_1) = y_3\}$$

or

$$\{x_2 \leq_2 y_2 \text{ for } f'(x_2) = x_3 \text{ and } f'(y_2) = y_3\}$$

\leq_3 is a partial order, as it is reflexive, antisymmetric and transitive and f' and g' are order-preserving maps by construction.

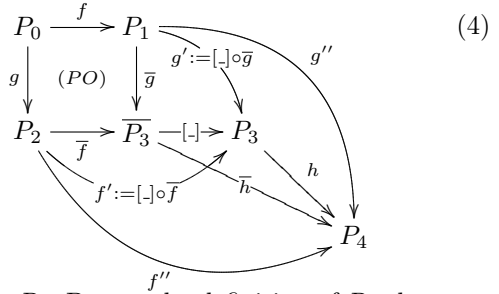
⁴ For example, let $P_0 = \{0, 5\}$ and $P_1 = \{0, 3, 5\}$ with f the inclusion and $P_2 = \{\bullet\}$, then $3 \leq_1 5$ yields $([3], [\bullet]) \in R_3$ and $0 \leq_1 3$ yields $([\bullet], [3]) \in R_3$, but $[\bullet] = \{0, 5\} \neq \{3\} = [3]$.

Subsequently we prove that $(P_1, \leq_1) \xrightarrow{g'} (P_3, \leq_3) \xleftarrow{f'} (P_2, \leq_2)$ is the pushout of $(P_1, \leq_1) \xleftarrow{f} (P_0, \leq_0) \xrightarrow{g} (P_2, \leq_2)$ in the category of partially ordered sets **PoS**ets:

Obviously $g' \circ f = f' \circ g$.

Given a partially ordered set (P_4, \leq_4) with $g'' \circ f = f'' \circ g$ in **PoS**ets, then we have $\bar{h} : \bar{P}_3 \rightarrow P_4$ in **S**ets due to the pushout (PO) in the diagram to the right. So, we define $h : P_3 \rightarrow P_4$ with $h([x]) = \bar{h}(x)$.

To prove that h is well-defined we show $h([x_3]) = h([y_3])$ with $x_3 \neq y_3$ and with $[y_3] = [x_3]$. Since $[y_3] = [x_3]$ and $x_3 \neq y_3$ there is $(x_3, y_3) \in \bar{R}_3$ and so $(x_3, y_3) \in R_3$ and $(y_3, x_3) \in R_3$. Due to the definition of R_3 there are four cases:



1. $\exists x_1, y_1 \in P_1 : x_1 \leq_1 y_1 \wedge \bar{g}(x_1) = x_3 \wedge \bar{g}(y_1) = y_3$
 $\wedge \exists x_2, y_2 \in P_2 : y_2 \leq_2 x_2 \wedge \bar{f}(x_2) = x_3 \wedge \bar{f}(y_2) = y_3$:
 Then we have $g''(x_1) = \bar{h} \circ \bar{g}(x_1) = \bar{h} \circ \bar{f}(x_2) = f''(x_2)$ and $g''(y_1) = \bar{h} \circ \bar{g}(y_1) = \bar{h} \circ \bar{f}(y_2) = f''(y_2)$. This yields $g''(x_1) \leq_4 g''(y_1)$ and $g''(y_1) = f''(y_2) \geq_4 f''(x_2) = g''(x_1)$. Since \leq_4 is an antisymmetric we have $g''(x_1) = g''(y_1)$. Hence, we have $h([x_3]) = \bar{h}(x_3) = \bar{h} \circ \bar{g}(x_1) = g''(x_1) = g''(y_1) = \bar{h} \circ \bar{g}(y_1) = \bar{h}(y_3) = h([y_3])$.
2. $\exists x_1, y_1 \in P_1 : y_1 \leq_1 x_1 \wedge \bar{g}(x_1) = x_3 \wedge \bar{g}(y_1) = y_3$
 $\wedge \exists x_2, y_2 \in P_2 : x_2 \leq_2 y_2 \wedge \bar{f}(x_2) = x_3 \wedge \bar{f}(y_2) = y_3$ analogously.
3. $\exists x_1, y_1 \in P_1 : x_1 \leq_1 y_1 \wedge \bar{g}(x_1) = x_3 \wedge \bar{g}(y_1) = y_3$
 $\wedge \exists x'_1, y'_1 \in P_1 : y'_1 \leq_1 x'_1 \wedge \bar{g}(x'_1) = x_3 \wedge \bar{g}(y'_1) = y_3$:
 So, we have $\bar{g}(x_1) = x_3 = \bar{g}(x'_1)$ and $\bar{g}(y_1) = y_3 = \bar{g}(y'_1)$. and $x_1 \leq_1 y_1$ and $y'_1 \leq_1 x'_1$. This yields $g''(x_1) \leq_4 g''(y_1)$ and $g''(y_1) = g''(y'_1) \leq_4 g''(x'_1) = g''(x_1)$. Since \leq_4 is an antisymmetric we have $g''(x_1) = g''(y_1)$. Hence, we have $h([x_3]) = \bar{h}(x_3) = \bar{h} \circ \bar{g}(x_1) = g''(x_1) = g''(y_1) = \bar{h} \circ \bar{g}(y_1) = \bar{h}(y_3) = h([y_3])$.
4. $\exists x_2, y_2 \in P_2 : x_2 \leq_2 y_2 \wedge \bar{f}(x_2) = x_3 \wedge \bar{f}(y_2) = y_3$
 $\wedge \exists x'_2, y'_2 \in P_2 : y'_2 \leq_2 x'_2 \wedge \bar{f}(x'_2) = x_3 \wedge \bar{f}(y'_2) = y_3$ analogously.

Moreover, $h \circ g' = h \circ [-] \circ \bar{g} = \bar{h} \circ \bar{g} = g''$ and $h \circ f' = h \circ [-] \circ \bar{f} = \bar{h} \circ \bar{f} = f''$.

Next we introduce the subclass of monomorphisms \mathcal{M} . Monomorphisms in **PoS**ets are the injective order preserving maps [5] and order embeddings – those mappings that satisfy item 1 in Definition 5 – are regular monomorphisms [5].

Definition 5 (Class \mathcal{M}). *The class \mathcal{M} is given by the class of strict order embeddings, that are order preserving mappings $f : (P, \leq_P) \rightarrow (P', \leq_{P'})$ that additionally are*

1. *order reflecting: $x \leq_P y$ if and only if $f(x) \leq_{P'} f(y)$ for $x, y \in P$*

2. and order strict: for each $z' \in P'$ with $f(x) \leq_{P'} z' \leq_{P'} f(y)$ there exists some $z \in P$ with $f(z) = z'$ (and hence $x \leq_P z \leq_P y$).

Class \mathcal{M} leads to pushouts that are constructed the same way as in the category **Sets**, hence the right adjoint functor $G : \mathbf{PoSets} \rightarrow \mathbf{Sets}$ preserves pushouts.

Lemma 3 (\mathcal{M} -Pushouts in **PoSets).** *Given $(P_1, \leq_1) \xleftarrow{f} (P_0, \leq_0) \xrightarrow{g} (P_2, \leq_2)$ with $f \in \mathcal{M}$ then there is the pushout $(P_1, \leq_1) \xrightarrow{g'} (P_3, \leq_3) \xleftarrow{f'} (P_2, \leq_2)$, such that in **Sets** $P_1 \xrightarrow{g'} P_3 \xleftarrow{f'} P_2$ is the pushout of $P_1 \xleftarrow{f} P_0 \xrightarrow{g} P_2$. Moreover, \mathcal{M} is stable under pushouts.*

Proof. Obviously, the construction of $\overline{R_3}$ in the proof of Lemma 2 yields for $f \in \mathcal{M}$ that $\overline{R_3} = ID$ the identity relation. Hence, $\overline{P_3} = \overline{P_3}|_{\overline{R_3}} = P_3$.

Moreover, it is \mathcal{M} -stable:

For $f \in \mathcal{M}$ in Diagram 4 we know that f' is injective, as pushouts in **Sets** preserve monomorphisms, i.e., injective mappings, and f' is order-preserving by construction.

f' is an order embedding. For $x_2, y_2 \in P_2$ and $f'(x_2) \leq_3 f'(y_2)$ we have due to the construction of \leq_3 four cases:

1. There are $x_1, y_1 \in P_1$ with $x_1 \leq_1 y_1$ so that $g'(x_1) = f'(x_2)$ and $g'(y_1) = f'(y_2)$. Due to the pushout construction there are $x_0, y_0 \in P_0$ with $x_0 \leq_0 y_0$ so that $f(x_0) = x_1$ and $g(x_0) = x_2$ and $f(y_0) = y_1$ and $g(y_0) = y_2$. Since g is order preserving, we have $x_2 \leq_2 y_2$.
2. There is $x_2 \leq_2 y_2$.
3. There is $z_3 \in P_3$ with $f'(x_2) \leq_3 z_3 \leq_3 f'(y_2)$, so that there are $x_1 \leq_1 z_1$ with $g'(x_1) = f'(x_2)$ and $g'(z_1) = z_3$ and $z_2 \leq_2 y_2$ and $f'(z_2) = z_3$.
Due to the pushout construction there are $x_0, z_0 \in P_0$ with $x_0 \leq_0 z_0$ so that $f(x_0) = x_1$ and $g(x_0) = x_2$ and $f(z_0) = z_1$ and $g(z_0) = z_2$. Since g is order preserving, we have $x_2 \leq_2 z_2 \leq_2 y_2$.
4. There is $z_3 \in P_3$ with $f'(x_2) \leq_3 z_3 \leq_3 f'(y_2)$, so that there are $z_1 \leq_1 y_1$ with $g'(z_1) = f'(x_2)$ and $g'(y_1) = f'(y_2)$ and $g'(z_1) = z_3$ and $x_2 \leq_2 z_2$ and $f'(z_2) = z_3$ analogously.

f' is a strict order embedding:

Let be $x_2, y_2 \in P_2$ and $f'(x_2) \leq_3 z_3 \leq_3 f'(y_2)$ given for $z_3 \in P_3$. Either $z_3 \in f'(P_2)$ and hence there is $f'(z_2) = z_3$ with $x_2 \leq_2 y_2$ or $z_3 \notin f'(P_2)$. Then there are $x_1, y_1, z_1, z'_1 \in P_1$ with $g'(x_1) = f'(x_2)$ and $g'(y_1) = f'(y_2)$ and $g'(z_1) = z_3 = g(z_1)$ and $x_1 \leq_1 z_1$ and $z'_1 \leq_1 y_1$. Due to the pushout construction there are $x_0, y_0 \in P_0$ with $f(x_0) = x_1$ and $g(x_0) = x_2$ and $f(y_0) = y_1$ and $g(y_0) = y_2$. Since f is a strict order embedding we have additionally, z_0, z'_0 with $f(z_0) = z_1$ and $f(z'_0) = z'_1$ and $x_0 \leq_0 z_0 \leq_0 z'_0 \leq_0 y_0$. Due to pushout construction $g(z_0) = g(z'_0)$ and as g is order preserving we have $x_2 = g(x_0) \leq_2 g(z_0) \leq_2 g(z'_0) = g(y_0) = y_2$ with $f'(g(z_0)) = z_3$.

Next we investigate pullbacks in **PoSets**.

Lemma 4 (Pullbacks in **PoSets).** *Given $(P_1, \leq_1) \xrightarrow{g} (P_0, \leq_0) \xleftarrow{f} (P_2, \leq_2)$ then there is the pullback $(P_1, \leq_1) \xleftarrow{f'} (P_3, \leq_3) \xrightarrow{g'} (P_2, \leq_2)$. Moreover, \mathcal{M} is stable under pullbacks.*

Proof. There is the pullback $P_1 \xleftarrow{f'} P_3 \xrightarrow{g'} P_2$ of $P_1 \xrightarrow{g} P_0 \xleftarrow{f} P_2$ in **Sets**.

$(P_1, \leq_1) \xleftarrow{f'} (P_3, \leq_3) \xrightarrow{g'} (P_2, \leq_2)$ with the partial order – given by $x_3 \leq_3 y_3$ if and only if $f'(x_3) \leq_1 f'(y_3)$ and $g'(x_3) \leq_1 g'(y_3)$ – is pullback in **PoSSets**. Obviously, f' and g' are order-preserving mappings.

\mathcal{M} -morphisms are monomorphisms and hence are preserved by pullbacks.

Theorem 1 (PoSets is an \mathcal{M} -Adhesive Category).

Proof. 1. The class \mathcal{M} in **PoSets** is PO-PB compatible, since

- pushouts along \mathcal{M} -morphisms exist and \mathcal{M} is stable under pushouts,
- pullbacks along \mathcal{M} -morphisms exist and \mathcal{M} is stable under pullbacks and
- obviously, \mathcal{M} contains all identities and is closed under composition.

2. In **PoSets** pushouts along \mathcal{M} -morphisms are \mathcal{M} -VK squares: In **PoSets** let be given a pushout as (1) in Definition 2 with $m \in \mathcal{M}$ and some commutative cube as (2) in Definition 2 with (1) being the bottom square and the back faces being pullbacks, then we have:

\Rightarrow : Let the top of (2) in Definition 2 be a pushout in **PoSets**. Pullbacks preserve \mathcal{M} -morphisms, so $m' \in \mathcal{M}$ and hence the top square is a pushout in **Sets** as well. As the category **Sets** is \mathcal{M} -adhesive, the front faces are pullbacks in **Sets** as well. Since the construction of pullbacks coincides in **Sets** and **PoSets**, the front faces are pullbacks in **PoSets**.

\Leftarrow : Let the front faces be pullbacks in **PoSets**, and hence pullbacks in **Sets**. Since $m \in \mathcal{M}$ (1) in Definition 2 is pushout in **Sets** as well. So, **Sets** being adhesive, we have the top square being a pushout in **Sets**. Moreover, $m' \in \mathcal{M}$ as the back face is a pullback preserving \mathcal{M} -morphisms. So, the top is a pushout along \mathcal{M} in **PoSets**.

Hence, **(PoSets, \mathcal{M})** is an \mathcal{M} -adhesive category.

Next we use Theorem 1 to prove that place/transition nets with transition priorities yield an \mathcal{M} -adhesive category. The definition of priorities allows a partial prioritisation. If several transitions have the highest priority, then the choice is again non-deterministic.

Definition 6. *The category of place/transition nets with transition priorities **PTp** is given by*

- *PT nets $N = (P, (T, \leq_T), pre, post, M_0)$ with $pre, post : G(T, \leq_T) \rightarrow P^\oplus$ for G defined in Lemma 1 and*
- *net morphisms $f = (f_P, f_T) : N_1 \rightarrow N_2$ where f_P is a mapping and f_T is an order-preserving map.*

A transition $t \in T$ is enabled under a marking M if $pre(t) \geq M$ and if there is no t' being enabled under M so that $t \leq t'$.

In the following we assume that \mathcal{M} is the class of net morphisms where f_P is strict and f_T is a strict order embedding.

Theorem 2 (($\mathbf{PTp}, \mathcal{M}$) is an \mathcal{M} -adhesive category).

Proof. The proof applies the construction for weak adhesive HLR categories (see Theorem 1 in [27]): We know that $(\mathbf{Sets}, \mathcal{M})$ with \mathcal{M} being the injective mappings is an \mathcal{M} -adhesive category and that $(-)^{\oplus} : \mathbf{Sets} \rightarrow \mathbf{Sets}$ preserves pullbacks along injective morphisms. As shown above $(\mathbf{PoSets}, \mathcal{M})$ with \mathcal{M} being the strict order embeddings is an \mathcal{M} -adhesive category and that $G : \mathbf{PoSets} \rightarrow \mathbf{Sets}$ preserves pushouts along \mathcal{M} -morphisms. So, the category \mathbf{PTp} is isomorphic to the comma category $ComCat(G, (-)^{\oplus}; I)$ with $I = 1, 2$, where $G : \mathbf{PoSets} \rightarrow \mathbf{Sets}$ is the right adjoint (see Lemma 1) from partial ordered sets to sets and $(-)^{\oplus}$ is the free commutative monoid functor and hence an \mathcal{M} -adhesive category.

Lemma 5 (The category of decorated PT nets with priorities is an \mathcal{M} -adhesive category).

Proof. See [23]; similar to the proof of Theorem 2 using \mathbf{decoPT} instead of \mathbf{PT} as the basis.

Lemma 6 (The category of AHL nets with priorities is an \mathcal{M} -adhesive category).

In [23] the result has been formulated in terms of abstract Petri nets [21], so that these extensions are valid for other types of Petri. In Example 3.5.1 in [21] it has been shown that AHL nets are an instantiation of abstract Petri nets.

6 Inhibitor Arcs

We now introduce inhibitor arcs [15], that allow places to inhibit a transition's firing. To that purpose a transition is mapped to its set of inhibiting places. So, inhibitor arcs are given as a function from transitions to the powerset of places.

Definition 7 (Inhibitor arcs). Given a place/transition net $N = (P, T, pre, post, M_0)$ inhibitor arcs are given by $inh : T \rightarrow \mathcal{P}(P)$.

A transition is then enabled under a marking M if additionally we have $M(p) = 0$ for all $p \in inh(t)$.

Lemma 7. The category of place/transition (PT) nets with inhibitor arcs \mathbf{PTi} is an \mathcal{M} -adhesive category with \mathcal{M} being the class of strict net morphisms.

Proof. The proof applies the construction for weak adhesive HLR categories (see Theorem 1 in [27]): Constructing the category \mathbf{PTi} using comma categories, we use the functor $F : \mathbf{PT} \rightarrow \mathbf{Sets}$ yielding the transition set T and the power set functor $\mathcal{P} : \mathbf{Sets} \rightarrow \mathbf{Sets}$. The category of PT nets is an \mathcal{M} -adhesive category (see [7]). Then the comma category $\mathbf{PTi} := ComCat(F, \mathcal{P}, \{inh\})$ yields the category of PT nets with inhibitor arcs and is a weak adhesive category as F preserves pushouts and \mathcal{P} pullbacks of injective morphisms. Hence, we have an \mathcal{M} -adhesive category, see [10].

Remark 2. Decorated place/transition nets with inhibitor arcs and algebraic high-level nets with inhibitor arcs also yield \mathcal{M} -adhesive categories (see [23]). Since the proofs of Theorem 2 and Lemma 7 are independent of each other, we can also obtain reconfigurable Petri nets with both inhibitor arcs and priorities.

7 Related Work

The focus of this section covers control structures in \mathcal{M} -adhesive transformation systems and Petri nets. In [24] a survey over control structures for reconfigurable Petri nets is given.

The control structures introduced in this paper for reconfigurable Petri nets have been introduced first for Petri nets, as labels, names, capacities. Inhibitor arcs as defined in [15] as well as priorities [2, 30] are well-known concepts in Petri nets. In contrast to [2, 30] where priorities are based on a mapping to the natural number, we define merely a partial order on the transitions. Inhibitor arcs in [4, 15] are defined as a relation $I \subseteq P \times T$, but this is equivalent to Definition 7 in Sect. 6. Changing transition labels allow the coordination between firing of transitions and application of rules [22].

Control structures in \mathcal{M} -adhesive transformation systems are required to specify the application of the rules more precisely. These control structures may determine the application of rules. They concern the situation that may or may not be given or they concern the order of the rules to be applied, namely net transformation units, rule priorities and application conditions. Negative application conditions have been formulated in terms of \mathcal{M} -adhesive transformation systems in [25]. Negative application conditions (NAC) for reconfigurable Petri nets have been introduced in [28] and provide the possibility to forbid certain rule applications. These conditions restrict the application of a rule forbidding a certain structure to be present before or after applying a rule in a certain context. Such a constraint influences thus each rule application or transformation and therefore changes significantly the properties of the net transformation system. A substantial extension of negative application conditions providing a much greater expressiveness are nested application conditions [8, 9, 19] that have been given in the framework of \mathcal{M} -adhesive transformation systems.

Graph transformation units have been introduced to graph grammars as the basic units for graph programming [1]. Control conditions can be given by regular expressions, describing in which order and how often the rules and imported units are to be applied. A large body of results has been developed since then [3, 13, 14], see also [17]. The formulation in terms of \mathcal{M} -adhesive transformation systems yields an abstract formulation of transformation units in terms of category theory [3].

8 Conclusion

In this paper we have introduced new control structures to reconfigurable Petri nets. In our case study the need to order the travel agents actions triggered the

use of priorities for transitions. That is a well-known concept in Petri nets, but has not been available for reconfigurable Petri nets. To obtain all the results of \mathcal{M} -adhesive transformation system for reconfigurable Petri nets with priorities we need to ensure that the corresponding category is again \mathcal{M} -adhesive. Moreover, we have shown that Petri nets with inhibitor arcs yield an \mathcal{M} -adhesive category. We have given the proofs in terms of place/transition nets and have argued that the results are valid for other kinds of Petri nets as well. These results allow a better control and hence a simplified and more precise modelling with reconfigurable Petri nets.

Future work concerns the realization of these concepts for decorated Petri nets in the tool RECONNET [6]. RECONNET provides possibilities to edit, to simulate and to verify reconfigurable Petri nets. The introduction of control structures has obviously a strong impact on these operations and needs to be integrated into the existing implementation. Ongoing work is the implementation of negative application conditions, so that control structures for the transformation part soon become available. The next task is the realization of transition priorities as well as inhibitor arcs.

References

1. Andries, M., Engels, G., Habel, A., Hoffmann, B., Kreowski, H., Kuske, S., Plump, D., Schürr, A., Taentzer, G.: Graph transformation for specification and programming. *Sci. Comput. Program.* **34**(1), 1–54 (1999)
2. Best, E., Koutny, M.: Petri net semantics of priority systems. *Theor. Comput. Sci.* **96**(1), 175–215 (1992)
3. Bottoni, P., Hoffmann, K., Parisi-Presicce, F., Taentzer, G.: High-level replacement units and their termination properties. *J. Vis. Lang. Comput.* **16**(6), 485–507 (2005)
4. Busi, N.: Analysis issues in petri nets with inhibitor arcs. *Theor. Comput. Sci.* **275**(1–2), 127–177 (2002)
5. Codara, P.: A theory of partitions of partially ordered sets. Ph.D. thesis, Università degli Studi die Milano (2007)
6. Ede, M., Hoffmann, K., Oelker, G., Padberg, J.: Reconnet: a tool for modeling and simulating with reconfigurable place/transition nets. *Electronic Communications of the EASST* **54**, 10 (2012)
7. Ehrig, H., Ehrig, K., Prange, U., Taentzer, G.: Fundamentals of Algebraic Graph Transformation. EATCS Monographs in TCS. Springer, Heidelberg (2006)
8. Ehrig, H., Golas, U., Habel, A., Lambers, L., Orejas, F.: M-Adhesive transformation systems with nested application conditions. part 2: embedding, critical pairs and local confluence. *Fundam. Inform.* **118**(1–2), 35–63 (2012)
9. Ehrig, H., Golas, U., Habel, A., Lambers, L., Orejas, F.: \mathcal{M} -adhesive transformation systems with nested application conditions. part 1: parallelism, concurrency and amalgamation. *Mathematical Structures in Computer Science* **24**(4), 48 (2014)
10. Ehrig, H., Golas, U., Hermann, F.: Categorical frameworks for graph transformation and HLR systems based on the DPO approach. *Bull. EATCS* **102**, 111–121 (2010)

11. Ehrig, H., Hoffmann, K., Padberg, J., Prange, U., Ermel, C.: Independence of net transformations and token firing in reconfigurable place/transition systems. In: Kleijn, J., Yakovlev, A. (eds.) ICATPN 2007. LNCS, vol. 4546, pp. 104–123. Springer, Heidelberg (2007)
12. Ehrig, H., Padberg, J.: Graph grammars and petri net transformations. In: Desel, J., Reisig, W., Rozenberg, G. (eds.) Lectures on Concurrency and Petri Nets. LNCS, vol. 3098, pp. 496–536. Springer, Heidelberg (2004)
13. Ermiler, M., Kreowski, H.-J., Kuske, S., von Totth, C.: From graph transformation units via minisat to grgen.net. In: Schürr, A., Varró, D., Varró, G. (eds.) AGTIVE 2011. LNCS, vol. 7233, pp. 153–168. Springer, Heidelberg (2012)
14. Hölscher, K., Klempien-Hinrichs, R., Knirsch, P.: Undecidable control conditions in graph transformation units. *Electron. Notes Theor. Comput. Sci.* **195**, 95–111 (2008)
15. Janicki, R., Koutny, M.: Semantics of inhibitor nets. *Inf. Comput.* **123**(1), 1–16 (1995)
16. Kahloul, L., Chaoui, A., Djouani, K.: Modeling and analysis of reconfigurable systems using flexible Petri nets. In: 4th IEEE International Symposium on Theoretical Aspects of Software Engineering, pp. 107–116 (2010)
17. Kreowski, H.-J., Kuske, S., Rozenberg, G.: Graph transformation units – an overview. In: Degano, P., De Nicola, R., Meseguer, J. (eds.) Concurrency, Graphs and Models. LNCS, vol. 5065, pp. 57–75. Springer, Heidelberg (2008)
18. Lack, S., Sobocinski, P.: Adhesive and quasiadhesive categories. *ITA* **39**(3), 511–545 (2005)
19. Lambers, L.: Certifying rule-based models using graph transformation. Ph.D. thesis, Berlin Institute of Technology (2009)
20. Llorens, M., Oliver, J.: Structural and dynamic changes in concurrent systems: reconfigurable Petri nets. *IEEE Trans. Comput.* **53**(9), 1147–1158 (2004)
21. Padberg, J.: Abstract Petri nets: a uniform approach and rule-based refinement. Ph.D. thesis, Technical University Berlin, Shaker Verlag (1996)
22. Padberg, J.: Abstract interleaving semantics for reconfigurable Petri nets. *Electron. Commun. EASST* **51**, 1–14 (2012)
23. Padberg, J.: Reconfigurable decorated PT nets with inhibitor arcs and transition priorities. CoRR abs/1409.6856 (2014). <http://arxiv.org/abs/1409.6856>
24. Padberg, J., Hoffmann, K.: A survey of control structures for reconfigurable Petri nets. *J. Comput. Commun.* **3**(2), 20–28 (2015)
25. Prange, U.: Towards algebraic high-level systems as weak adhesive HLR categories. *Electron. Notes Theor. Comput. Sci.* **203**(6), 67–88 (2008)
26. Prange, U., Ehrig, H., Hoffmann, K., Padberg, J.: Transformations in reconfigurable place/transition systems. In: Degano, P., De Nicola, R., Meseguer, J. (eds.) Concurrency, Graphs and Models. LNCS, vol. 5065, pp. 96–113. Springer, Heidelberg (2008)
27. Prange, U., Ehrig, H., Lambers, L.: Construction and properties of adhesive and weak adhesive high-level replacement categories. *Appl. Categorical Struct.* **16**(3), 365–388 (2008)
28. Rein, A., Prange, U., Lambers, L., Hoffmann, K., Padberg, J.: Negative application conditions for reconfigurable place/transition systems. *Electron. Commun. EASST* **10**, 1–14 (2008)
29. Reiseverband, D.: Fakten und Zahlen 2013 zum deutschen Reisemarkt (2013). https://www.driv.de/fileadmin/user_upload/Fachbereiche/Statistik_und_Marktforschung/Fakten_und_Zahlen/14-03-17/_DRV/_Zahlen/_Fakten2013/_V2.pdf, last visited: 03/24/2015 15:54

30. Werner, M., Popova-Zeugmann, L., Richling, J.: A method to prove non-reachability in priority duration Petri nets. *Fundam. Inform.* **61**(3–4), 351–368 (2004)
31. www.amadeus.com: The global travel ecosystem: a more personalized traveler journey (2014). <http://www.amadeus.com/media/130by2020/index.html#>, last visited: 03/17/2015 12:58