# A Taste of Categorical Petri Nets

Claudia Ermel, Alfio Martini

**Bericht-Nr. 96-9**

# A Taste of Categorical Petri Nets

Claudia Ermel Alfio Martini

e-mail:{alfio,lieske}@cs.tu-berlin.de

**Abstract**

This report aims at providing introductory concepts for a categorical approach for the study of Petri Nets. After motivating why a categorical approach for studying petri nets might be desirable, we show that "classical" place/transition Nets, usually seen as bipartite directed graphs, can be naturally given a monoid structure. Upon this idea we construct a category with place/transition Nets as objects and a suitable notion of place/transition Net morphisms as morphisms. We also verify the existence of ubiquitous categorical constructions in order to verify issues concerning cocompleteness. In the end, the usual notion of semantics of place/transition Nets by a marking graph construction is shown to be nothing else than an adjoint situation between two suitable functors. Further points of interest are pointed out in the bibliographic notes.

## Contents

# 1 Introduction

Petri nets are used to describe processes as concurrent and interacting machines which engage in internal actions and communications with their environment or user. They are the first model of concurrent systems which has been developed and, in their various evolutions, the most heavily used in many applications. Petri nets are also a well-known specification technique for concurrent and distributed systems, the reasons being its simplicity, universality, graphical representation and support of computer tools.

However, some main objections against Petri nets both as a model for concurrency and as a specification technique from a software engineering point of view can be summarized by saying that Petri nets lack abstraction, compositionality, data-type handling, refinement and structuring. Data-type handling is usually analyzed in connection with High-Level Nets. The other ones can be properly formulated and studied by bringing the language of category theory into Petri net theory. As a consequence, we are here concerned mainly with the link between category theory and Petri Net theory, that is to say, we want to provide here a taste of a collection of definitions and results concerning categorical Petri nets, and thus provide the reader with an adequate basis to follow the current literature dealing with questions such as structuring, abstraction, compositionality and refinement.

The slogan here is, after [8], "Petri nets are monoids", i.e., a place/transition net is simply an ordinary, directed graph equipped with two algebraic operations. Especially, a net provides the generators of the algebraic structure. This observation allows one straightforwardly to bring categorical methods into Petri net theory.

In the first section we review basic concepts of place transition nets needed for this presentation with the sole purpose of fixing notation and terminology used throughout. In the second section we introduce categorical Petri nets, the equivalence between the classical and the categorical formulation and a discussion about the natural representation of concurrent transition in the categorical approach. In the last one, the usual marking graph semantical construction for place/transition nets is shown to be nothing but a free construction within the categorical terminology.

## 2 Preliminaries

The only purpose of this section is to present basic concepts related to place/transition nets which we will need for further development of the material contained in this chapter. In case the reader is already familiar with them, a quick look might suffice so as to fix the notation used throughout.

Place/transition nets are composed of two basic components: a set of places and a set of transitions. Besides, it is necessary to define a relationship between places and transitions. Later on, we will need to considerer a distribution of tokens (a primitive concept) in the set of places, called marking, which has the purpose of controlling the execution of a place/transition net. These observations lead directly to the following

**Definition 2.1** *A 4-tuple $N =_{def} (P, T, F, W)$ is called a place/transition-net (P/T-Net) where*

1. *$P$ and $T$ are disjunct sets, whose elements are called* **places** *and* **transitions***, respectively.*
2. *$F \subseteq (P \times T) \cup (T \times P)$ is a binary relation, the* **flow relation** *of $N$.*
3. *$W : F \to \mathbb{N} \setminus \{0\}$ is a mapping which associates to each edge of the net its* **weight***.*

**Remark 2.2** The usual definition of a P/T-Net requires imposing a (possibly) limited capacity to each place. We consider here that any place has an unlimited capacity, since every net with limited place capacity can uniquely be extended through the addition of the so-called complementary places to a net of an unlimited place capacity (see, e.g, [14]).

The above definition of the flow relation may suggest that a P/T-Net may be viewed as a bipartite directed graph, where an edge is directed from a place $p_i$ to a transition $t_j$ if $\langle p_i, t_j \rangle \in F$ and conversely, there is an edge directed from $t_j$ to $p_i$ if $\langle t_j, p_i \rangle \in F$.

Before turning to the dynamic properties of a P/T-Net, we need the following auxiliary

**Definition 2.3** *Let $N$ be P/T-Net. For $x \in P \cup T$, $\bullet x =_{def} \{y | \langle y, x \rangle \in F\}$ is called the* **pre-condition** *of $x$, and $x\bullet =_{def} \{y | \langle x, y \rangle \in F\}$ is called the* **post-condition** *of $x$.*

**Remark 2.4** Note that a P/T-Net is a straightforward generalization of a finite state machine if we restrict the sets $\bullet x$ and $x\bullet$ for all $x \in T$ to be singletons and set to each edge of the net, the constant weight 1.

**Example 2.5** In the P/T-Net of Figure 1 (see 2.8) we have $\bullet t_2 = \{p_2, p_6\}, p_1\bullet = \{t_1, t_4\}$, and so on.

In addition to the static properties represented by a P/T-Net graph, a P/T-Net has dynamic properties that result from its execution, which is controlled by the position and movement of tokens. The use of tokens rather resembles a board game. The tokens are moved by the *firing* of the transitions. A transition $t$ must be *enabled* in order to fire. It is enabled when each place $p$ belonging to the set of its pre-condition has at least as many tokens as defined by the weight of the edge connecting $p$ and $t$. The transition fires by removing $W(p,t)$ tokens from each place $p$ of the pre-condition and producing new tokens by depositing in each place $p$ belonging to the post-condition set of $t$ the number of tokens determined by the edge connecting each $p$ with $t$. More formally, we have the following

**Definition 2.6 (Marking and firing)** *Let $N$ be a place/transition net. Then*

1. *A mapping $M : P \to \mathbb{N}$ is called a* **marking** *on $N$.*
2. *A transition $t \in T$ is M-enabled if and only if $\forall p \in \bullet t : M(p) \geq W(p,t)$.*
3. *A $M$-enabled transition $t \in T$ may yield a* **follower marking** *$M'$ of $M$ which is such that for each $p \in P$*

$$M'(p) =_{def} \begin{cases} M(p) - W(p,t) & \text{if } p \in \bullet t - t\bullet \\ M(p) + W(t,p) & \text{if } p \in t\bullet - \bullet t \\ M(p) - W(p,t) + W(p,t) & \text{if } p \in \bullet t \cap t\bullet \\ M(p) & \text{otherwise} \end{cases}$$

*We say $t$* **fires from** *$M$* **to** *$M'$, and we write $M[t\rangle M'$.*

**Example 2.7** Considering $M_0 : P \to \mathbb{N}$ as in Figure 1. Then $M_0[t_1\rangle M_1$ is given by:

$$\begin{aligned} M_1(p_1) &= M_0(p_1) - W(p_1,t_1) = 3 - 1 = 2, \\ M_1(p_2) &= M_0(p_2) + W(t_1,p_2) = 0 + 1 = 1, \\ M_1(p_3) &= M_0(p_3) = 0, \\ M_1(p_4) &= M_0(p_4) = 0, \\ M_1(p_5) &= M_0(p_5) = 0, \\ M_1(p_6) &= M_0(p_6) = 0. \end{aligned}$$

**Example 2.8 (Readers-Writers Problem)**. Figure 1 shows a P/T-Net illustrating the readers-writers problem. There are several variants of the readers-writers problem, but the basic structure is the same. Processes are of two types: readers processes and writer processes. All processes share a common file, variable, or data object. Reader processes never modify the object, while writer processes do modify it. Thus writer processes must mutually exclude all other reader and writer processes, but multiple reader processes can access the shared data simultaneously. The problem is to define a control structure which does not deadlock or allow violations of the mutual exclusion criteria. Figure 1 depicts a solution where the number of processes is bounded by 3 (i.e., tokens are seen as processes), where in our graphical representation of P/T-Nets places are represented by circles and transitions by boxes. The edges $f \in F$ are labeled by $W(f)$ if $W(f) > 1$, and a marking $M$ is represented by drawing $M(p)$ tokens on each place $p \in P$.

The formal description of this net, according to 2.1, can be given as follows:

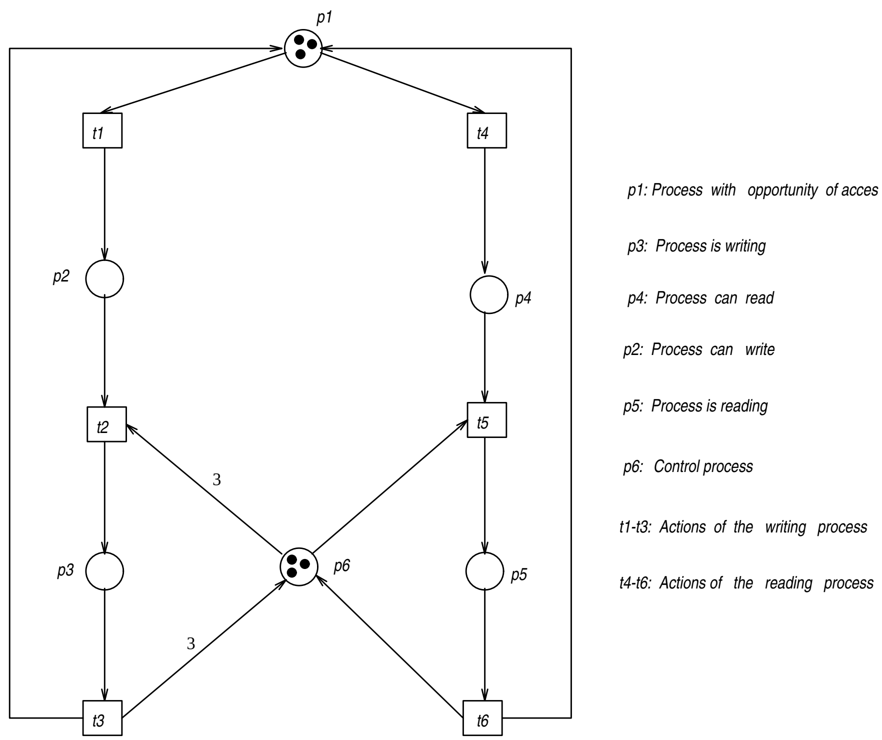


Figure 1: P/T-Net for a reader-writer system

$$
\begin{array}{rcl}
P & = & \{p_1, p_2, p_3, p_4, p_6\}, \\
T & = & \{t_1, t_2, t_3, t_4, t_5, t_6, \} \\
F & = & \{\langle p_1, t_1\rangle, \langle t_1, p_2\rangle, \langle p_2, t_2\rangle, \langle t_2, p_3\rangle, \langle p_3, t_3\rangle, \langle t_3, p_1\rangle, \langle p_1, t_4\rangle, \\
 & & \langle t_4, p_4\rangle, \langle p_4, t_5\rangle, \langle t_5, p_5\rangle, \langle p_5, t_6\rangle, \langle t_6, p_1\rangle, \langle p_6, t_2\rangle, \langle p_6, t_5\rangle, \\
 & & \langle t_3, p_6\rangle, \langle t_6, p_6\rangle\}, \\
W & : & W(\langle p_1, t_1\rangle) = 1, W(\langle t_1, p_2\rangle) = 1, \ldots, W(\langle p_6, t_2\rangle) = 3, \\
 & & W(\langle p_6, t_5\rangle) = 1, W(\langle t_3, p_6\rangle) = 3, W(\langle t_6, p_6\rangle) = 1, \\
M_0 & : & M_0(p_1) = M_0(p_6) = 3, M_0(p_2) = M_0(p_3) = M_0(p_4) = M_0(p_5) = 0
\end{array}
$$

Two transitions are independently enabled, when the firing of one transition does not prevent the firing of the other, or in other words, each place belonging to the intersection of the pre-condition set of the two transitions contains "enough' tokens for the occurrence of both firings. Otherwise the enabled transitions are in conflict. More precisely, we have:

**Definition 2.9** *Let $N$ be a place/transition net, $t_1, t_2 \in T$ two transitions from $N$, and let $M$ be a marking from $N$. Then, $t_1, t_2$ are said to be* **independently enabled** *when the following holds:*

$$\forall p \in \bullet t_1 \cap \bullet t_2 : M(p) \geq W(p, t_1) + W(p, t_2).$$

*Otherwise, $t_1$ and $t_2$ are said to be in* **conflict**.

**Example 2.10** In the P/T-Net of Figure 1 the transitions $t_1$ and $t_4$ are independently enabled. By the parallel firing of the two transitions, we get a resulting marking where the transitions $t_2$ and $t_5$ are enabled. However, at this time, the two enabled transitions are in conflict, since the firing of one necessarily prevents the firing of the other (as desired).

In such independent situations, in which the "concrete" sequence of firings plays no role in the resulting system's state, it is reasonable to suggest that the corresponding transitions might execute in parallel. It is exactly in this sense that Petri nets model parallel or concurrent systems. However, these systems may have properties which are not present in sequential ones. For instance, due to parallelism, the system can get into a "deadlock" situation, i.e., a situation in which no other transition can fire because two or more processes simultaneously prevent the transition of one another. In Petri nets, the concept of deadlock is normally studied in connection with the concept of *liveness*.

**Example 2.11** As an example of a deadlock situation, consider Figure 2 where two processes share two resources. In a first step, process 1 need one resource (the token on place $P_{b1}$) and process 2 the other one (the token on place $P_{b2}$). In a second step, process 1 needs the resource that lies on $P_{b2}$ and deposits the first resource again on $P_{b1}$. On the other hand, process 2 needs the resource from $P_{b1}$ and deposits the second on $P_{b2}$. The third step initializes the system to its initial state (initial marking). The system runs correctly whenever the two processes execute sequentially. A complication arises when the both independently enabled transitions $T_1$ and $T_4$ fire in parallel or one after the other. In such a situation, no transition can fire again and a deadlock arises.

From a marking $M$, a set of transition firings is possible. The result of firing a transition in a marking $M$ is a new marking $M'$, where in this case, $M'$ is a immediately reachable marking from $M$, resulting from firing a enabled transition in $M$. We can also say that a marking $M'$ is reachable from $M$ if it is immediately reachable from $M$ or it is reachable from any marking that it is immediately reachable from $M$. If we take the reflexive, transitive closure of this "immediate reachable" relation we get the reachability set of a marked P/T-Net. This is formally represented in the following

**Definition 2.12** *Given a P/T-Net with a marking $M$, the reachable set $[M\rangle$, is the smallest set of markings such that*

- *$M \in [M\rangle$ and*
- *if $M_1 \in [M\rangle$ and for some $t \in T$ $M_1[t\rangle M_2$ then $M_2 \in [M\rangle$.*

A marking gives the actual system's state of a net. The marking graph of a P/T-Net represents all possible markings, i.e., all permissible system's states of the P/T-Net together with all possible firing transitions between these states, and hence can be seen as the semantics of a P/T-Net.

**Definition 2.13** *The* **marking graph** $MG =_{def} (MG_V, MG_E, s^{MG}, t^{MG})$ *of a P/T-Net is given by:*

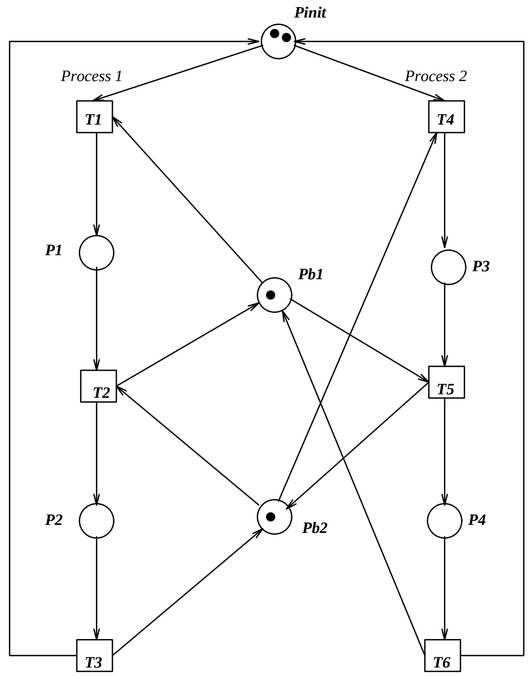


Figure 2: P/T-Net for a deadlock situation

$$\begin{aligned}
MG_V &=_{def} \{M | M : P \to \mathbb{N} \textit{ is a marking for } N\}, \\
MG_E &=_{def} \{(M, t, M') | t \in T \wedge M[t\rangle M', M \in MG_V\}, \\
s^{MG}(M, t, M') &=_{def} M, \\
t^{MG}(M, t, M') &=_{def} M'.
\end{aligned}$$

Since the marking graph contains all markings and all possible transitions, we have also a number of markings and corresponding transitions which have no sense from the point of view of the modeled system. In the case of our reader-writer system, the only markings which have sense are those markings in which the sum of the tokens in the places $p_1, p_2, p_3, p_4$ and $p_5$ is exactly equal to three, since otherwise the principle of mutual exclusion would not work. This motivate us to present the following

**Definition 2.14** *A* **marked** *P/T-Net* $N_{M_0} =_{def} (P, T, F, W, M_0)$ *is a P/T-Net* $N =_{def} (P, T, F, W)$ *together with a marking* $M_0 : P \to \mathbb{N}$*, which will be called* **initial marking**.

Besides, when marked nets are treated, we are often not interested in the whole set of possible transitions, but only on those ones which are possible taking into account the initial marking. This information give us the reachability graph.

The vertices of the reachability graph of a P/T-Net $N$ are obtained from the reachable set of markings from the initial marking $M_0$ (see 2.12). The corresponding edges are given by the directed edges which have as sources (or targets) markings from $[M_0\rangle$.

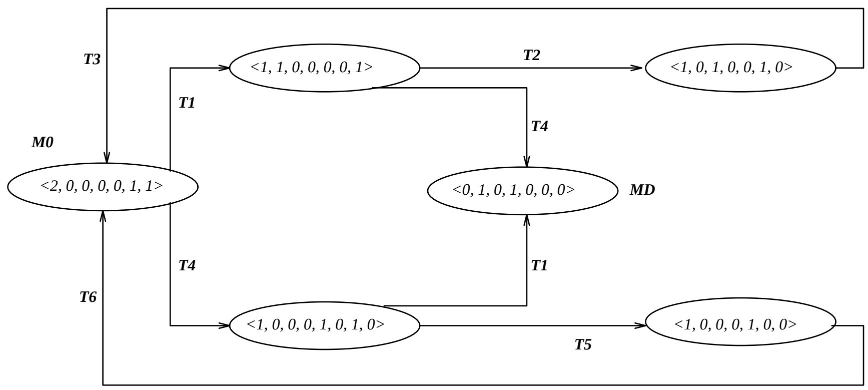


Figure 3: Reachability graph for the P/T-Net of Figure 2

**Definition 2.15** *The* **sequential reachability** *graph* $EG =_{def} (EG_V, EG_E, s^{EG}, S^{EG})$ *of a marked P/T-net* $N_{M_0}$ *is given by*

$$\begin{aligned} EG_V &=_{def} [M_0\rangle, \\ EG_E &=_{def} \{(M, t, M' | M[t\rangle M' \text{ with } M, M' \in [M_0\rangle, t \in T\}, \\ s^{EG}(M, t, M') &=_{def} M, \\ t^{EG}(M, t, M') &=_{def} M'. \end{aligned}$$

**Remark 2.16** Note that the sequential reachability graph is in fact a subgraph of marking graph of 2.13.

**Example 2.17** We present in Figure 3 the reachability graph from the P/T-Net of Figure 2, where the marking of the places is expressed in tuple form as $\langle M(Pinit), M(P_1), M(P_2), M(P_3), M(P_4), M(P_{b1}), M(P_{b2})\rangle$. It is easy to see that the marking $M_D$ represents a deadlock situation, since it is not the source of any edge. Besides, such a state is obtained after the firing of $T_1$ and $T_4$ (in any sequence).

# 3 Categorical P/T-Nets

## 3.1 Motivation

Petri nets are able to distinguish clearly the basic concepts in the behavior of processes. The graphical representation of nets visualizes these concepts and are also suitable for modeling of very detailed, machine-like view of processes. However, there are at least two serious problems with Petri nets:

- **Compositionality**. Missing is a convenient way of composing or decomposing larger nets from or into smaller ones by a set of high level operators.
- **Abstraction**. The concept of Petri nets is a low-level concept, comparable with machine level programming language. However, in specification of complex systems, there

is a need to abstract from internal transitions and focus on the communication behavior between processes, i.e., we may need sometimes to treat processes as "black-boxes" in the same way we treat modules in programming languages.

On the other hand, the application of categorical methods in connection to P/T-Nets has the following advantages when comparing with the classical, more established approach:

- The firing of a unique transition as well as the concurrent firing of various transitions can be homogeneously treated and described.
- By bringing in P/T-Nets into the mathematical context of category theory, we have at our disposal the whole toolkit of categorical methods and concepts which can usually be immediately applied when approaching questions concerning structuring, compositionality and refinement.
- The possibility of comparing and relating Petri nets with other models for concurrent systems, like event-structures, transition systems, synchronization trees, etc., by giving to each one of them a categorical structure and using well-known categorical concepts like functors, natural transformations and adjoint situations in order to establish semantical relations amongst these models.

In the following we introduce elementary concepts concerning monoids and show that the usual concept of Petri nets can be given a "monoid structure".

## 3.2 Petri nets are monoids

Before we motivate a justification for the title of this section, we need the following auxiliary

**Definition 3.2.1** *Let $P$ be a set. Then $\mathcal{P} =_{def} (P^*, \lambda, \oplus)$ is called the* **free monoid generated** *by $P$, where $P^*$ is the set of all words over $P$, such that for all $u, v, w \in P^*$ the following equations hold:*

- $v \oplus \lambda = \lambda \oplus v = \lambda$,
- $u \oplus (v \oplus w) = (u \oplus v) \oplus w$,

*that is to say, $\oplus$ is a associative binary operation in $\mathcal{P}$ with identity $\lambda$.*

**Remark 3.2.2**

1. $P^*$ is also called the the **Kleene closure** over $P$.
2. If $\mathcal{P}$ is a free commutative monoid, i.e., if for all $w, v \in P^*$, $w \oplus v = v \oplus w$, then any word of $P^*$ can be represented as a linear sum. For instance, if $P =_{def} \{a, b, c, d\}$, then $a \oplus b \oplus c \oplus a \oplus c \oplus d \oplus c = a \oplus a \oplus b \oplus c \oplus c \oplus c \oplus d \in P^*$ can be represented as "$2a \oplus b \oplus 3c \oplus d$". In this case, we will stick to the notation $P^{\oplus}$ in order to denote the **free commutative monoid** generated by $P$.

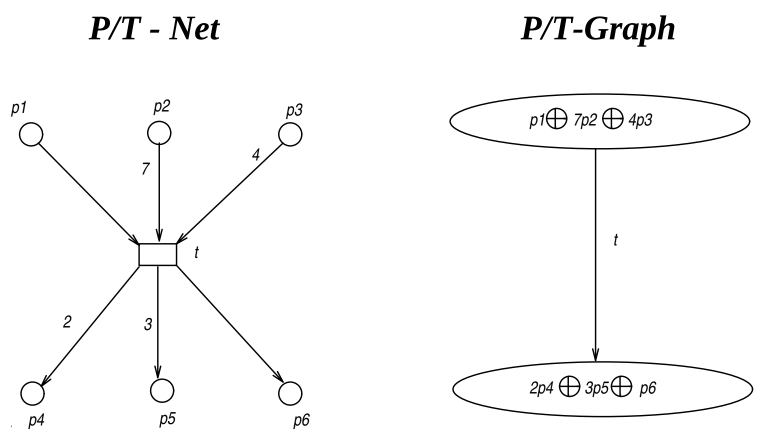


Figure 4: Unmarked P/T-Net

3. In general, we have the following formal definition for any $w \in P^{\oplus}$ expressed in linear sum form:

$$w =_{def} \sum_{i=1}^{n} k_i a_i, \; k_i \in \mathbb{N}, a_i \in P.$$

4. Commutative monoids together with monoid homomorphisms define a subcategory **CMon** of **Mon**, the latter one being the category of monoids and monoid homomorphisms.

Consider now the "unmarked" P/T-Net on the left of Figure 4. It has a set of places $P =_{def} \{p_1, p_2, p_3, p_4, p_5, p_6\}$. The main point is that this P/T-Net can be understood as a ordinary graph whose set of vertices are elements of the base set of the free commutative monoid $P^{\oplus}$ over $P$ in the following way:

- The places $p_1, p_2, p_3$ belonging to the pre-condition of $t$ together with their respective weights must described as an element of $P^{\oplus}$, i.e., $p_1 \oplus 7p_2 \oplus 4p_3$. This is the first vertex of the graph.
- Likewise, the places belonging to the post-condition of $t$ together with their respective weights must also be described as an element of $P^{\oplus}$, i.e., $2p_4 \oplus 3p_5 \oplus p_6$. This is the second vertex of the graph.
- The resulting graph is obtained by an edge labeled $t$ whose source is the first vertex and whose target is the second vertex, as described above. The resultant graph is depicted on the right of Figure 4. Such a graph will be called P/T-Graph from now on.

Based on the above considerations we can now give a monoid structure to our classical P/T-Net, which is expressed in the following

**Definition 3.2.3** *Given a (classical) P/T-Net* $N =_{def} (P, T, F, W)$, *a* **categorical P/T-Net** $\mathcal{N} =_{def} (P, P^{\oplus}, T, pre, post : T \to P^{\oplus})$ *is a graph with a set of vertices* $P^{\oplus}$, *a set of edges* $T$, *as well the pre- and post-condition mappings (source and target mappings, respectively)* $pre$ *and* $post$, *respectively.*

**Remarks 3.2.4**

1. We will use interchangeably the terms "categorical P/T-Net" and "P/T-Graph".
2. A P/T-Graph usually contain countably infinite isolated vertices (all the elements of $P^{\oplus}$ which don't correspond to any pre- or post-condition of transitions in the P/T-Net).
3. For each transition $t$ in the P/T-Net, there is a corresponding edge in the P/T-Graph with label $t$ and where the mappings $pre(t), post(t)$ are defined.
4. For every transition without pre-condition, rep. without post-condition, the corresponding edge of the P/T-Graph has source, resp. target, the $\lambda$-vertex.

**Example 3.2.5** As a more substantial example, let us now consider our readers/writers system from Figure 1 without the initial marking. The corresponding categorical P/T-Net $\mathcal{N}$ is defined by the mappings $pre, post$ in the following way:

$$
\begin{array}{ll}
pre(t_1) = p_1 & post(t_1) = p_2 \\
pre(t_4) = p_1 & post(t_1) = p_4 \\
pre(t_2) = p_2 \oplus 3p_6 & post(t_2) = p_3 \\
pre(t_3) = p_3 & post(t_3) = 3p_6 \oplus p_1 \\
pre(t_5) = p_4 \oplus p_6 & post(t_5) = p_5 \\
pre(t_6) = p_5 & post(t_6) = p_6 \oplus p_1
\end{array}
$$

The corresponding P/T-Graph is depicted in Figure 5 (note that isolated vertices suggest the "infinity" of the set of vertices in $P^{\oplus}$).

The above remarks and example demand the following

**Theorem 3.2.6** *Classical P/T-Nets and categorical Petri nets are in one-to-one correspondence to each other.*

**Proof:** First we show that for each classical P/T-Net $N =_{def} (P, T, F, W)$, there is a categorical P/T-Net $\mathcal{N} =_{def} (P, P^{\oplus}, T, pre, post)$ with $pre, post : T \to P^{\oplus}$. Now, for all $t \in T, pre, post$ are defined as:

$$
pre(t) =_{def} \begin{cases} \sum_{\langle p,t\rangle \in F} W(p,t) \cdot p & \text{if } \exists p \in P : \langle p,t\rangle \in F \\ \lambda & otherwise \end{cases}
$$

$$
post(t) =_{def} \begin{cases} \sum_{\langle t,p\rangle \in F} W(t,p) \cdot p & \text{if } \exists p \in P : \langle t,p\rangle \in F \\ \lambda & otherwise \end{cases}
$$

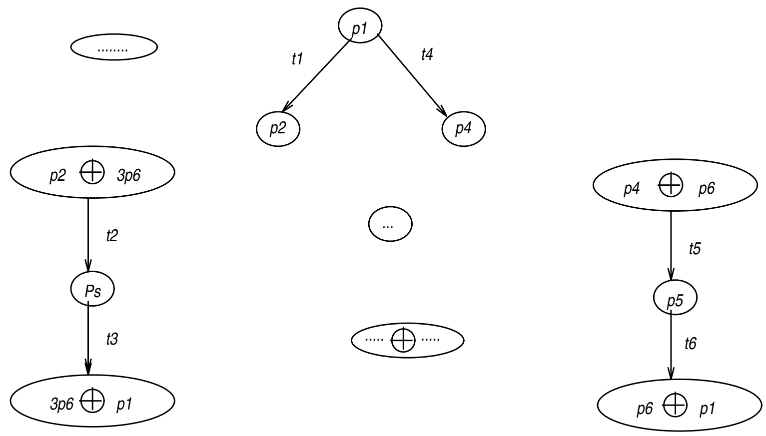


Figure 5: P/T-Graph for the readers/writers P/T-Net

Now, to each categorical P/T-Net $\mathcal{N} =_{def} (P, P^{\oplus}, T, pre, post)$ with $pre, post : T \to P^{\oplus}$, there is a corresponding classical P/T-Net $N =_{def} (P, T, F, W)$, where the flow relation $F$ and the weight $W$ are defined as follows:

$$F =_{def} \{\langle p, t\rangle | t \in T \wedge pre(t) = \sum_{i=1}^{n} k_i p_i \wedge \exists j \in \{1, \ldots, n\} : p = p_j \wedge k_j \neq 0\}$$

$$\cup \{\langle t, p\rangle | t \in T \wedge post(t) = \sum_{i=1}^{n} k_i p_i \wedge \exists j \in \{1, \ldots, n\} : p = p_j \wedge k_j \neq 0\}$$

$$W(\langle p, t\rangle) =_{def} k_j, \text{ if } pre(t) = \sum_{i=1}^{n} k_i p_i \wedge \exists j \in \{1, \ldots, n\} : p = p_j \wedge k_j \neq 0\}$$

$$W(\langle t, p\rangle) =_{def} k_j, \text{ if } pre(t) = \sum_{i=1}^{n} k_i p_i \wedge \exists j \in \{1, \ldots, n\} : p = p_j \wedge k_j \neq 0\}$$

$\square$

**Remark 3.2.7** Due to the above equivalence, it is not necessary to differentiate the qualifying terms "classical" and "categorical".

Now consider the P/T-Net in Figure 6. If we conceive the marking of a P/T-Net also as a linear sum from $P^{\oplus}$, then we have $M = 3pz \oplus 2pl$, and we can describe the firing of a Petri net also in an algebraic way. This is reflected in the following

**Definition 3.2.8** *A transition $t \in T$ is enabled under a marking $M$, denoted $M[t\rangle$, when $pre(t) \leq M$ (where $\leq$ is defined componentwise on the summands of the linear sum).*

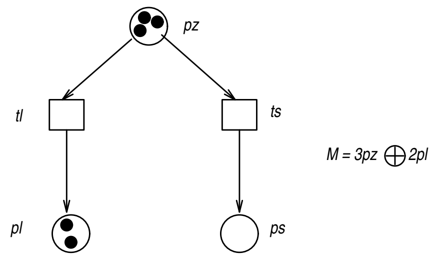


Figure 6: A marked P/T-Net

### 3.3 The category P/T-Net

**Definition 3.3.1** *Let $\mathcal{N}_i =_{def} (P_i^{\oplus}, T_i, pre_i, post_i : T_i \rightarrow P_i^{\oplus}), i = 1, 2$ be P/T-Nets. Then a P/T-Net morphism $f : \mathcal{N}_1 \rightarrow \mathcal{N}_2 =_{def} \langle f_P, f_T \rangle$ is a pair of mappings $f_P : P_1 \rightarrow P_2, f_T : T_1 \rightarrow T_2$, such that the following diagram commutes componentwise:*

$$\begin{tikzcd} T_1 \arrow[r, shift left, "pre_1"] \arrow[r, shift right, "pre_2"'] \arrow[d, "f_T"'] & P_1^{\oplus} \arrow[d, "f_P^{\oplus}"] \\ T_2 \arrow[r, shift left, "post_1"] \arrow[r, shift right, "post_2"'] & P_2^{\oplus} \end{tikzcd}$$

*where $f_P^{\oplus}$ is the free extension of $f_P$, i.e., for every $w = p_1 \ldots p_n \in P_1^{\oplus}, n \geq 0$,*

$$f_P^{\oplus}(w) = f_P(p_1) \ldots f_P(p_n)$$

**Remark 3.3.2** The free extension above must be a monoid homomorphism, i.e., the following equations should necessarily hold for all $w_1, w_2 \in P_1^{\oplus}$:

$$\begin{aligned} f^{\oplus}(\lambda) &= \lambda \\ f^{\oplus}(w_1 \oplus w_2) &= f^{\oplus}(w_1) \oplus f^{\oplus}(w_2) \end{aligned}$$

**Example 3.3.3** Consider the two P/T-Nets in Figure 7 together with their associated P/T-Graphs. Then $f_P =_{def} \{p_1 \mapsto p_5, p_2 \mapsto p_5, p_3 \mapsto p_6, p_4 \mapsto p_6\}$ and $f_T =_{def} \{t_1 \mapsto t_2\}$ is a P/T-Net morphism according to 3.3.1.

With the concept of P/T-morphism available, we can now introduce the desired

**Proposition 3.3.4** *The class of all P/T-Nets $\mathcal{N} =_{def} (P, P^{\oplus}, T, pre, post : T \rightarrow P^{\oplus})$ together with the set of all P/T-Net morphisms define a category* **P/T-Net**.

**Proof:** First, given P/T-Nets $\mathcal{N}_i, i = 1, \ldots, 3$ according to 3.2.3 and corresponding $P/T$-net morphisms $f : \mathcal{N}_1 \rightarrow \mathcal{N}_2, g : \mathcal{N}_2 \rightarrow \mathcal{N}_3$ according to 3.3.1, we have to show that $g \circ f : N_1 \rightarrow N_3$ is also a P/T-Net morphism. We put $\langle (g \circ f)_P, (g \circ f)_T \rangle =_{def} \langle g_P \circ f_P, g_T \circ f_T \rangle$. This means

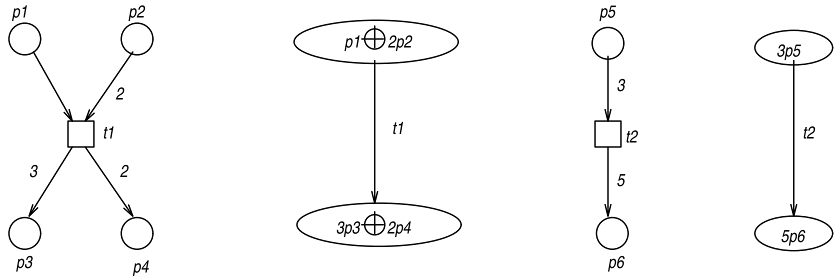


Figure 7: An example of a P/T-Net morphism

that we have to show that the following diagram commutes (note that here we are already using the fact of the existence of a free functor $(-)^{\oplus} : \mathbf{Set} \to \mathbf{CMon}$ of A.6.6, such that $(g_P \circ f_P)^{\oplus} = g_P^{\oplus} \circ f_P^{\oplus}$):

$$\begin{tikzcd}
T_1 \arrow[r, shift left, "pre_1"] \arrow[r, shift right, "post_1"'] \arrow[d, "f_T"'] & P_1^{\oplus} \arrow[d, "f_P^{\oplus}"] \\
T_2 \arrow[r, shift left, "pre_2"] \arrow[r, shift right, "post_2"'] \arrow[d, "g_T"'] & P_2^{\oplus} \arrow[d, "g_P^{\oplus}"] \\
T_3 \arrow[r, shift left, "pre_3"] \arrow[r, shift right, "post_3"'] & P_3^{\oplus}
\end{tikzcd}$$

But by assumption, the two squares commute, and hence the whole diagram commutes.

Now, given any P/T-Net $\mathcal{N}_1 =_{def} (P_1, P_1^{\oplus}, T_1, pre_1, post_1 : T_1 \to P_1^{\oplus})$, we define the identity on $\mathcal{N}_1$ as $id_{\mathcal{N}_1} =_{def} \langle id_P : P_1 \to P_1, id_T : T_1 \to T_1 \rangle$ such that this pair becomes a P/T-Net morphism. Now, for any $f : \mathcal{N}_1 \to \mathcal{N}_2$, we have that $f \circ id_{\mathcal{N}_1} = id_{\mathcal{N}_1}$ and $id_{\mathcal{N}_2} \circ f = f$, since these equalities hold componentwise for the base mappings, which are just morphisms in **Set**.

It remains to show associativity of composition, i.e., given morphisms $f : \mathcal{N}_1 \to \mathcal{N}_2, g : \mathcal{N}_2 \to \mathcal{N}_3, h : \mathcal{N}_3 \to \mathcal{N}_4$, we have to show that $h \circ (g \circ f) = (h \circ g) \circ f$. But this equality again holds trivially, since componentwise composition is associative because the base mappings are morphisms in **Set**. $\square$

## 3.4 Colimits in P/T-Net

We now turn our attention to the internal structure of the category **P/T-Net**, but with special attention on how to appropriately compose P/T-Nets. Since structuring concepts in specification theory (e.g., [4],[3]) can be suitable modeled by colimit constructions, we ask the natural question whether **P/T-Net** is colimit complete. According to A.5.9, we have to show that **P/T-Net** has products and coequalizers. This is nothing but the task of our next two propositions.

**Proposition 3.4.1 P/T-Net** *has coproducts.*

**Proof:** Let $\mathcal{N}_i =_{def} (P_i, P_i^{\oplus}, T_i, pre_i, post_i : T_i \to P_i^{\oplus}), i = 1, 2$ be P/T-Nets. We claim that $\mathcal{N}_1 + \mathcal{N}_2 =_{def} (P_1 \uplus P_2, (P_1 \uplus P_2)^{\oplus}, [pre_1, pre_2], [post_1, post_2] : T_1 \uplus T_2 \to (P_1 \uplus P_2)^{\oplus})$ is a coproduct in **P/T-Net** (where $\uplus$ is the disjoint union of sets).

First note that there exists a free functor $(-)^{\oplus}$ : **Set** $\to$ **CMon** (see A.6.6) such that we put $(-)^{\oplus}(P_1 \uplus P_2) = (P_1 \uplus P_2)^{\oplus}$, where $(P_1 \uplus P_2)^{\oplus} = P_1^{\oplus} + P_2^{\oplus}$, since $(-)^{\oplus}$, by A.6.5, preserves colimits and hence coproducts. Moreover, we define $[pre_1, pre_2]$ and $[post_1, post_2]$ in the following way

$$[pre_1, pre_2](t) =_{def} \begin{cases} pre_1(t) & \text{if } t \in T_1 \\ pre_2(t) & \text{if } t \in T_2 \end{cases}$$

$$[post_1, post_2](t) =_{def} \begin{cases} post_1(t) & \text{if } t \in T_1 \\ post_2(t) & \text{if } t \in T_2 \end{cases}$$

This shows that $\mathcal{N}_1 + \mathcal{N}_2$ is a well defined P/T-Graph. Now, in order to show that $\mathcal{N}_1 + \mathcal{N}_2$ is a coproduct in **CMon**, we have to show that for given P/T-Graph morphisms $f : \mathcal{N}_1 \to \mathcal{M}, g : \mathcal{N}_2 \to \mathcal{M}$, where $\mathcal{M} =_{def} (P_M, P_M^{\oplus}, T_M, pre_M, post_M : T \to P_M^{\oplus})$ there is a unique $h : \mathcal{N}_1 + \mathcal{N}_2 \to \mathcal{M}$ such that the following diagram commutes:

$$\begin{tikzcd} & \mathcal{M} & \\ \mathcal{N}_1 \arrow[ur, "f"] \arrow[r, "in_{\mathcal{N}_1}"'] & \mathcal{N}_1 + \mathcal{N}_2 \arrow[u, "h"] & \mathcal{N}_2 \arrow[ul, "g"'] \arrow[l, "in_{N_2}"] \end{tikzcd}$$

(1)

Now consider the following coproduct diagrams in **Set**.

$$\begin{tikzcd} & P_M & \\ P_1 \arrow[ur, "f_P"] \arrow[r, "in_{P_1}"'] & P_1 \uplus P_2 \arrow[u, "{[f_P, g_P]}"] & P_2 \arrow[ul, "g_P"'] \arrow[l, "in_{P_2}"] \end{tikzcd}$$

(2)

$$\begin{tikzcd} & T_M & \\ T_1 \arrow[ur, "f_T"] \arrow[r, "in_{T_1}"'] & T_1 \uplus T_2 \arrow[u, "{[f_T, g_T]}"] & T_2 \arrow[ul, "g_T"'] \arrow[l, "in_{T_2}"] \end{tikzcd}$$

(3)

Since the free functor $(-)^{\oplus}$ : **Set** $\to$ **CMon** preserves colimits we have the following commutative diagram in **CMon** (the translation of (2) along $(-)^{\oplus}$).

$$\begin{tikzcd} & P_M^{\oplus} & \\ P_1^{\oplus} \arrow[ur, "f_{P_1}^{\oplus}"] \arrow[r, "in_{P_1}^{\oplus}"'] & (P_1 \uplus P_2)^{\oplus} \arrow[u, "{[f_{P_1}^{\oplus}, g_{P_2}^{\oplus}]}"] & P_2^{\oplus} \arrow[ul, "g_{P_2}^{\oplus}"'] \arrow[l, "in_{P_2}^{\oplus}"] \end{tikzcd}$$

(3)

where $in^{\oplus}_{P_1} =_{def} (-)^{\oplus}(in_{P_1}), in^{\oplus}_{P_2} =_{def} (-)^{\oplus}(in_{P_2}), f^{\oplus}_P =_{def} (-)^{\oplus}(f_P), g^{\oplus}_P =_{def} (-)^{\oplus}(g_P)$.

We define $h = \langle h_P, h_T] : \mathcal{N}_1 + \mathcal{N}_2 \to \mathcal{N}$ were $h_P : P_1 \uplus P_2 \to P_M, h_T : T_1 \uplus T_2 \to T_M$ by $h =_{def} \langle [f_P, g_P], [f_T, g_T] \rangle$. To see that $h$ is indeed a morphism in **P/T-Net**, we have to show that the following diagram commutes:

$$\begin{tikzcd}
T_1 \uplus T_2 \arrow[r, shift left, "{[pre_1, pre_2]}"] \arrow[r, shift right, "{[post_1, post_2]}"'] \arrow[d, "{[f_T, g_T]}"'] & (P_1 \uplus P_2)^{\oplus} \arrow[d, "{[f^{\oplus}_P, g^{\oplus}_P]}"] \\
T_M \arrow[r, shift left, "pre_M"] \arrow[r, shift right, "post_M"'] & P_M
\end{tikzcd}$$

We distinguish two cases:

**Case** $t \in T_1$

$$\begin{array}{rcll}
[f^{\oplus}_P, g^{\oplus}_P] \circ [pre_1, pre_2](t) & = & [f^{\oplus}_P, g^{\oplus}_P](pre_1(t)) & \text{(by definition of } [pre_1, pre_2]) \\
& = & f^{\oplus}_P(pre_1(t)) & \text{(by definition of } [f^{\oplus}_P, g^{\oplus}_P]) \\
& = & pre_M(f_T(t)) & (f \text{ is a } \textbf{P/T-Graph} \text{ morphism}) \\
& = & pre_M \circ [f_T, g_T](t) & \text{(by definition of } [f_T, g_T])
\end{array}$$

**Case** $t \in T_2$:

$$\begin{array}{rcll}
[f^{\oplus}_P, g^{\oplus}_P] \circ [pre_1, pre_2](t) & = & [f^{\oplus}_P, g^{\oplus}_P](pre_e(t)) & \text{(by definition of } [pre_1, pre_2]) \\
& = & g^{\oplus}_P(pre_2(t)) & \text{(by definition of } [f^{\oplus}_P, g^{\oplus}_P]) \\
& = & pre_M(g_T(t)) & (g \text{ is a } \textbf{P/T-Graph} \text{ morphism}) \\
& = & pre_M \circ [f_T, g_T](t) & \text{(by definition of } [f_T, g_T])
\end{array}$$

Applying a similar argument we get $[f^{\oplus}_P, g^{\oplus}_P] \circ [post_1, post_2] = post_M \circ [f_T, g_T]$. This shows that the above diagram diagram commutes and hence that $h$ is a **P/T-Graph** morphism.

Now, in order to see that (1) commutes observe that

$$\begin{array}{rcll}
h \circ in_{N_1} & = & \langle [f_P, g_P], [f_T, g_T] \rangle \circ \langle in_{P_1}, in_{T_1} \rangle & \text{(by definition)} \\
& = & \langle [f_P, g_P] \circ in_{P_1}, [f_T, g_T] \circ in_{T_1} \rangle & \text{(by composition)} \\
& = & \langle f_P, f_T \rangle & \text{(by (2) and (3))} \\
& = & f & \text{(by definition)}
\end{array}$$

as well as

$$\begin{array}{rcll}
h \circ in_{N_2} & = & \langle [f_P, g_P], [f_T, g_T] \rangle \circ \langle in_{P_2}, in_{T_2} \rangle & \text{(by definition)} \\
& = & \langle [f_P, g_P] \circ in_{P_2}, [f_T, g_T] \circ in_{T_2} \rangle & \text{(by composition)} \\
& = & \langle g_P, g_T \rangle & \text{(by (2) and (3))} \\
& = & g & \text{(by definition)}
\end{array}$$

It remains to show uniqueness of $h =_{def} \langle [f_P, g_P], [f_T, g_T] \rangle$. Therefore, suppose there is another $h' =_{def} \langle [f'_P, g'_P], [f'_T, g'_T] \rangle : \mathcal{N}_1 + \mathcal{N}_2 \to \mathcal{M}$ such that (4) commutes. But this implies, by the above calculations, that $[f'_P, g'_P]$ and $[f'_T, g'_T]$ would also make, respectively, (1) and

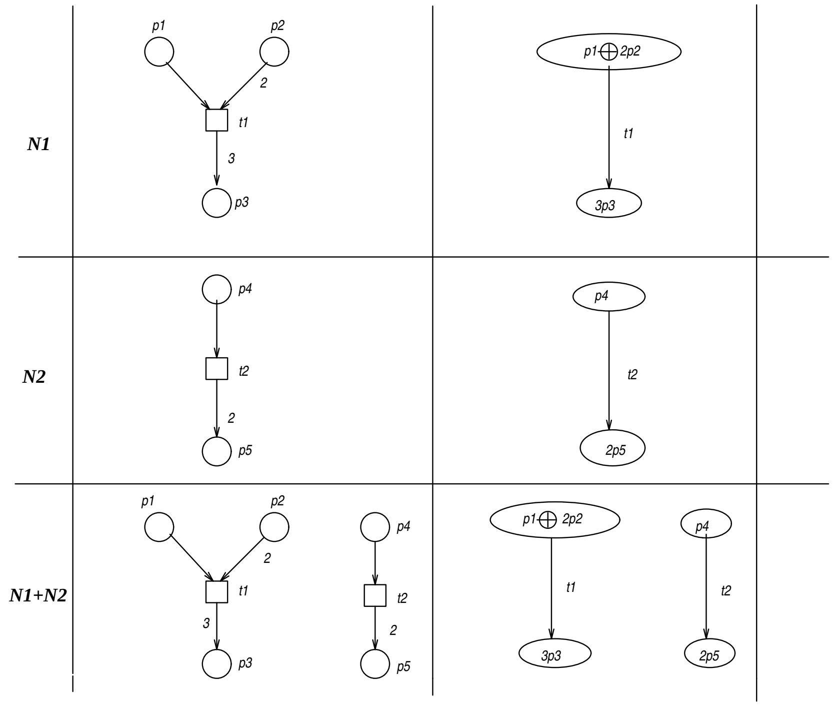


Figure 8: A coproduct of P/T-Nets

(2) commute. Now $[f_P, g_P]$ and $[f_T, g_T]$ are unique in (2) and (3) by the couniversal property of coproducts and therefore, $[f_P, g_P] = [f'_P, g'_P]$ as well as $[f_T, g_T] = [f'_T, g'_T]$, i.e., $h = h'$, as required.

$\square$

**Example 3.4.2** Figure 8 shows an example of a coproduct from two P/T-Nets $\mathcal{N}_1$ and $\mathcal{N}_2$. Note that the coproduct $\mathcal{N}_1 + \mathcal{N}_2$ of two P/T-Nets is the result of a composition operation without synchronization, i.e., both nets are put together without any kind of cooperation.

**Proposition 3.4.3** *The category* **P/T-Net** *has coequalizers.*

**Proof:** Given two P/T-Net morphisms $f, g : \mathcal{N}_1 \to \mathcal{N}_2$ we must define a P/T-Net $\mathcal{N}_C$ and a morphism $c : \mathcal{N}_2 \to \mathcal{N}_C$ with $c \circ f = c \circ g$ such that for any other $\mathcal{N}_D$ and $d : \mathcal{N}_2 \to \mathcal{N}_D$ with $d \circ f = d \circ g$, there is a unique $h : \mathcal{N}_C \to \mathcal{N}_D$ such that $h \circ c = d$ according to the following diagram:

$$\begin{tikzcd}
\mathcal{N}_1 \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & \mathcal{N}_2 \arrow[r, "c"] \arrow[dr, "d"'] & \mathcal{N}_C \arrow[d, "h"] \\
 & & \mathcal{N}_D
\end{tikzcd}$$

(1)

Now consider the following diagram, where the coequalizer $c_T$ is defined by the following diagram in **Set** (knowing that **Set** is cocomplete):

$$\begin{tikzcd} T_1 \arrow[r, shift left, "f_T"] \arrow[r, shift right, "g_T"'] & T_2 \arrow[r, "c_T"] \arrow[dr, "d_T"'] & T_C \arrow[d, "h_T"] \\ & & T_D \end{tikzcd}$$

(2)

where $T_C =_{def} T_2/_{\equiv}$, with $\equiv$ the equivalence relation induced by $\{\langle f_T(t), g_T(t)\rangle | t \in T_1\}$, that is to say, $c_T : T_2 \to T_C$ is the natural mapping $t \mapsto [t]$.

Now, consider the equalizer $c_P : P_2 \to P_C$ in the category **Set** according to the following diagram:

$$\begin{tikzcd} P_1 \arrow[r, shift left, "f_P"] \arrow[r, shift right, "g_P"'] & P_2 \arrow[r, "c_P"] \arrow[dr, "d_P"'] & P_C \arrow[d, "h_P"] \\ & & P_D \end{tikzcd}$$

(3)

where again $P_C =_{def} P_2/_{\equiv}$, with $\equiv$ the equivalence relation induced by $\{\langle f_P(p), g_P(p)\rangle | p \in P_1\}$, that is to say, $c_P : P_2 \to P_C$ is the natural mapping $p \mapsto [p]$.

Since there is a free functor $(-)^{\oplus} : \textbf{Set} \to \textbf{CMon}$, we define $P^{\oplus} =_{def} (-)^{\oplus}(P_C), c_P^{\oplus} =_{def} (-)^{\oplus}(c_P)$ and obtain $c_P^{\oplus}$ as a coequalizer in **CMon** according to the following diagram (since $(-)^{\oplus}$ preserves colimits and hence coequalizers):

$$\begin{tikzcd} P_1^{\oplus} \arrow[r, shift left, "f_{P^\oplus}"] \arrow[r, shift right, "g_{P^\oplus}"'] & P_2^{\oplus} \arrow[r, "c_{P^\oplus}"] \arrow[dr, "d_{P^\oplus}"'] & P_C^{\oplus} \arrow[d, "h_{P^\oplus}"] \\ & & P_D^{\oplus} \end{tikzcd}$$

(4)

Now consider the following diagram:

$$
\begin{tikzcd}
T \arrow[r, shift left, "pre_1"] \arrow[r, shift right, "post_1"'] \arrow[d, shift right, "f_T"'] \arrow[d, shift left, "g_T"] & P^{\oplus} \arrow[d, shift right, "g_P^{\oplus}"'] \arrow[d, shift left, "f_P^{\oplus}"] \\
T_2 \arrow[r, shift left, "pre_2"] \arrow[r, shift right, "post_2"'] \arrow[d, "c_T"'] & P^{\oplus} \arrow[d, "c_P^{\oplus}"] \\
T_C \arrow[d, "h_T"'] & P_C^{\oplus} \arrow[d, "h_P^{\oplus}"] \\
T_D \arrow[r, shift left, "pre_D"] \arrow[r, shift right, "post_D"'] & P_D^{\oplus}
\end{tikzcd}
$$

where, by assumption, we have that $c_P^{\oplus}$ is a coequalizer of $g_P^{\oplus}, f_P^{\oplus}$ and $c_T$ a coequalizer of $f_T, g_T$. But now note that

$$
\begin{aligned}
c_P^{\oplus} \circ pre_2 \circ f_T &= c_P^{\oplus} \circ f_P^{\oplus} \circ pre && (f \text{ is a P/T-morphism}) \\
&= c_P^{\oplus} \circ g_P^{\oplus} \circ pre && (c_P^{\oplus} \text{ is a coequalizer}) \\
&= c_P^{\oplus} \circ pre_2 \circ g_T && (g \text{ is a P/T-morphism})
\end{aligned}
$$

This shows that $(c_P^{\oplus} \circ pre_2) \circ f_T = (c_P^{\oplus} \circ pre_2) \circ g_T$. But, by assumption, $c_T$ is the coequalizer of $f_T, g_T$, and hence, there must be a unique $pre_C : T_C \to P_C^{\oplus}$ such that $pre_C \circ c_T = c_P^{\oplus} \circ pre_2$.

By a similar argument, there is a unique $post_C : T_C \to P_C^{\oplus}$ such that $post_C \circ c_T = c_P^{\oplus} \circ post_2$. This implies the commutativity of the middle square in the following diagram.

$$
\begin{tikzcd}
T_1 \arrow[r, shift left, "pre_1"] \arrow[r, shift right, "post_1"'] \arrow[d, shift right, "f_T"'] \arrow[d, shift left, "g_T"] & P_1^{\oplus} \arrow[d, shift right, "g_P^{\oplus}"'] \arrow[d, shift left, "f_P^{\oplus}"] \\
T_2 \arrow[r, shift left, "pre_2"] \arrow[r, shift right, "post_2"'] \arrow[d, "c_T"'] & P_2^{\oplus} \arrow[d, "c_P^{\oplus}"] \\
T_C \arrow[r, shift left, "pre_C"] \arrow[r, shift right, "post_C"'] \arrow[d, "h_T"'] & P_C^{\oplus} \arrow[d, "h_P^{\oplus}"] \\
T_D \arrow[r, shift left, "pre_D"] \arrow[r, shift right, "post_D"'] & P_D^{\oplus}
\end{tikzcd}
$$

Now we put $\mathcal{N}_C =_{def} (P_C^{\oplus}, T_C, pre_C, post_C : T_C \to P_C^{\oplus})$, which is by the above arguments a well-defined P/T-Net. Moreover, the commutativity of the middle square above asserts that $\langle c_P, c_T \rangle : \mathcal{N}_2 \to \mathcal{N}_C$ becomes a P/T-Net morphism. We must now show that the bottom square also commutes so that also $h$ becomes a P/T-Net morphism. To see this, note that

$$
\begin{aligned}
pre_D \circ h_T \circ c_T &= d_T \circ pre_D && (\text{by } (2)) \\
&= d_P^{\oplus} \circ pre_2 && (d \text{ is a P/T-Net morphism}) \\
&= h_P^{\oplus} \circ c_P^{\oplus} \circ pre_2 && (\text{by } (4)) \\
&= h_P^{\oplus} \circ pre_C \circ c_T && (c \text{ is a P/T-Net morphism})
\end{aligned}
$$

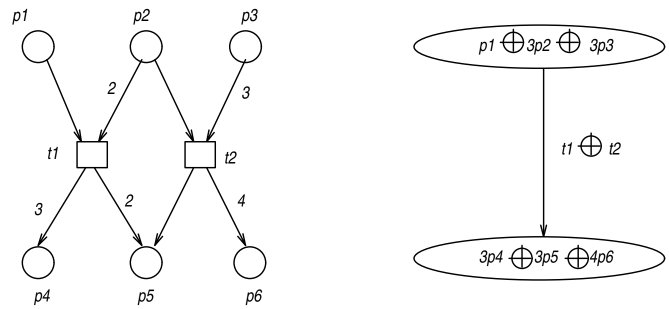


Figure 9: Parallel firing in a P/T-Graph

Now, since $c_T$ is a coequalizer, it is a epimorphism (see A.3.7), and hence $(pre_D \circ h_T) \circ c_T = (h_P^{\oplus} \circ pre_C) \circ c_T \Rightarrow pre_D \circ h_T = h_P^{\oplus} \circ pre_C$ which is the desired equality to show that $h =_{def} \langle h_P, h_T \rangle$ is a P/T-Net morphism. It still remains to show equality of (1) and uniqueness of $h$. To see that (1) commutes note that

$$\begin{array}{rcll} h \circ c & = & \langle h_P, h_T \rangle \circ \langle c_P, c_T \rangle & \text{(by definition)} \\ & = & \langle h_P \circ c_P, h_T \circ c_T \rangle & \text{(composition)} \\ & = & \langle d_P, d_T \rangle & (c_P \text{ and } c_T \text{ are coequalizers}) \\ & = & d & \text{(by definition)} \end{array}$$

Finally, to see uniqueness of $h$, suppose that there is another $i =_{def} \langle i_P, i_T \rangle$ such that $i \circ c = d$. But this means that $\langle i_P, i_T \rangle \circ \langle c_P, c_T \rangle = \langle i_P \circ c_P, i_T \circ c_T \rangle = \langle d_P, d_T \rangle$, i.e., $i_P \circ c_P = d_P$ and $i_T \circ c_T = d_T$. But $h_T$ and $h_P$ are unique in (2) and (3) such that we must have, by uniqueness, $i_P = h_P$ and $i_T = h_T$, that is to say, $i = h$. □

# 4 Free constructions and the semantics of categorical P/T-Nets

## 4.1 Parallel firing of transitions

The parallel firing of several transitions in P/T-Nets can be naturally expressed in P/T-Graphs. Both the pre-condition as well as the post-condition of the firing transitions can be expressed as a common pre- or post-condition as elements of $P^{\oplus}$. At the same time, the simultaneous transition of more than one transition can also be expressed as a linear sum. For instance, in Figure 9, the possible parallel firing of $t_1$ and $t_2$ can be expressed by the corresponding P/T-Graph on the right. Therefore, a monoid structure is defined both for the vertices as well as for the edges of a P/T-Graph.

Obviously, the parallel firing of transitions can only be realized when they are enabled by the current marking. This leads to the next

**Definition 4.1.1** *Given a P/T-Net $\mathcal{N} =_{def} (P, P^{\oplus}, T, pre, post : T \to P^{\oplus})$, we say that two*

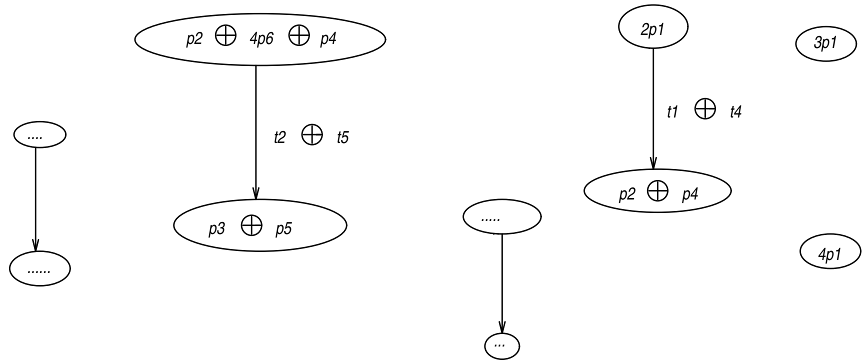


Figure 10: Subgraph of a parallel P/T-Graph for the reader-writer system

*transitions* $t_1, t_2 \in T$ *are* **independently enabled** *when* $pre(t_1) \oplus pre(t_2) \leq M$*. The parallel firing of the transitions* $t_1$ *and* $t_2$ *build a parallel firing step* $t_1 \oplus t_2$*.*

**Example 4.1.2** As an example, consider the readers-writers system of Figure 1. Under $M_0 = 3p_1 \oplus 3p_6$, the transitions $t_4$ and $t_1$ are independently enabled, since we have $pre(t_1) \oplus pre(t_4) = 2p_1 \leq M_0$. On the other hand, the transition $t_5$ and $t_2$ are not independently enabled for any marking in the reachable set $[M_0\rangle$, since $pre(t_5) \oplus pre(t_2) = p_2 \oplus 4p_6 \oplus p_4$ is bigger that every possible marking in $[M_0\rangle$ (since the number of tokens in $p6$ is never bigger than 3).

The above example motivates us to present the following

**Definition 4.1.3** *A* **free P/T-Monoid** $\mathcal{N}^\oplus =_{def} (P, P^\oplus, T^\oplus, pre^\oplus, post^\oplus : T^\oplus \rightarrow P^\oplus)$ *consists of a P/T-Net, such that the* $T^\oplus$ *is the free commutative monoid generated by* $T$*, and* $post^\oplus, post^\oplus : T^\oplus \rightarrow P^\oplus$ *are the unique extensions of* $pre, post : T \rightarrow P^\oplus$*, given for all* $t_1 \oplus \ldots \oplus t_n \in T^\oplus$ *by:*

$$pre^\oplus(t_1 \oplus \ldots \oplus t_n) = pre(t_1) \oplus \ldots \oplus pre(t_n),$$

*that is to say,* $pre^\oplus$ *and* $post^\oplus$ *are free monoid homomorphisms.*

**Remark 4.1.4** We speak simply about a (non-free) commutative monoid $\mathcal{T}$ when the monoid of transitions is not necessarily free. Thus, the pre- and post-conditions functions $pre, post : \mathcal{T} \rightarrow P^\oplus$ are simple monoid homomorphisms.

**Example 4.1.5** As an example of a subgraph of a parallel graph for our reader-writer system consider Figure 10. Note that additional subgraphs and infinitely many isolated vertices are only suggested. Moreover, a P/T-Net in monoid representation with parallel firing of transitions (called **parallel** P/T-Graph) contain all possible combination of transitions, since the current marking plays no role for the generation of the such a parallel graph.

**Definition 4.1.6** *Let* $\mathcal{N}_i^{\oplus} =_{def} (P_i, P_i^{\oplus}, T_i^{\oplus}, pre_i^{\oplus}, post_i^{\oplus} : T_i^{\oplus} \to P_i^{\oplus}), i = 1, 2$ *be free P/T-Monoids. A free P/T-Monoid morphism is a P/T-Net morphism* $f =_{def} \langle f_P, f_T \rangle$ *where* $f_P : P_1 \to P_2, f_T : T_1 \to T_2$, *and* $f_P^{\oplus} : P_1^{\oplus} \to P_2^{\oplus}, f_T^{\oplus} : T_1^{\oplus} \to T_2^{\oplus}$ *are free monoid homomorphisms, such that the following diagram commutes componentwise:*

$$\begin{tikzcd}
T_1^{\oplus} \arrow[r, shift left, "pre_1^{\oplus}"] \arrow[r, shift right, "post_1^{\oplus}"'] \arrow[d, "f_T^{\oplus}"'] & P_1^{\oplus} \arrow[d, "f_P^{\oplus}"] \\
T_2 \arrow[r, shift left, "pre_2^{\oplus}"] \arrow[r, shift right, "post_2^{\oplus}"'] & P_2^{\oplus}
\end{tikzcd}$$

With the available concepts of free P/T-Monoids and free P/T-Monoid morphisms we can establish our next

**Proposition 4.1.7** *The class of all free P/T-Monoids together with set of all free P/T-Monoid morphisms constitute a category, called* **FM-Graph**.

**Proof:** Let $\mathcal{N}_1^{\oplus}, \mathcal{N}_2^{\oplus}, \mathcal{N}_3^{\oplus}$ be free P/T-Monoids according to 4.1.3 and $f =_{def} \langle f_P, f_T \rangle : \mathcal{N}_1^{\oplus} \to \mathcal{N}_2^{\oplus}, g =_{def} \langle g_P, g_T \rangle : \mathcal{N}_2^{\oplus} \to \mathcal{N}_3^{\oplus}$ be free P/T-Monoid morphisms according to 4.1.6. To say that $g \circ f : \mathcal{N}_1^{\oplus} \to \mathcal{N}_3^{\oplus}$ is again a free P/T-Monoid morphism amounts to say that the following diagram is componentwise commutative,

$$\begin{tikzcd}
T_1^{\oplus} \arrow[r, shift left, "pre_1^{\oplus}"] \arrow[r, shift right, "post_1^{\oplus}"'] \arrow[d, "f_T^{\oplus}"'] & P_1^{\oplus} \arrow[d, "f_P^{\oplus}"] \\
T_2 \arrow[r, shift left, "pre_2^{\oplus}"] \arrow[r, shift right, "post_2^{\oplus}"'] \arrow[d, "g_T^{\oplus}"'] & P_2^{\oplus} \arrow[d, "g_P^{\oplus}"] \\
T_3 \arrow[r, shift left, "pre_3^{\oplus}"] \arrow[r, shift right, "post_3^{\oplus}"'] & P_3^{\oplus}
\end{tikzcd}$$

which follows directly from the commutativity of the two squares by assumption. Now for any free P/T-Monoid $\mathcal{N}_1^{\oplus}$ we have an identity P/T-Monoid morphism $id_{\mathcal{N}_1} =_{def} \langle id_P, id_T \rangle : \mathcal{N}_1^{\oplus} \to \mathcal{N}_1^{\oplus}$, such that for any P/T-Monoid morphism $f : \mathcal{N}_1 \to \mathcal{N}_2$ the equations $id_{\mathcal{N}_2} \circ f = f$ and $f \circ id_{\mathcal{N}_1}$ hold because they hold componentwise in **Set**. The same argument applies for the associativity axiom.

□

## 4.2 Reflexive transitions and the marking graph

We now turn our attention to construct a marking graph in the sense of 2.13 from our parallel P/T-Graph. If we examine more closely our parallel P/T-Graph, we will see that there are transitions which exist in a marking graph but are still missing in our parallel P/T-Graph.

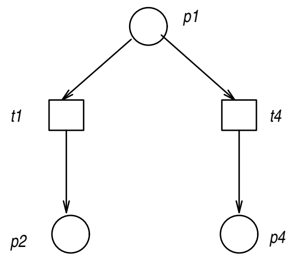


Figure 11: Subnet of the reader-writer system

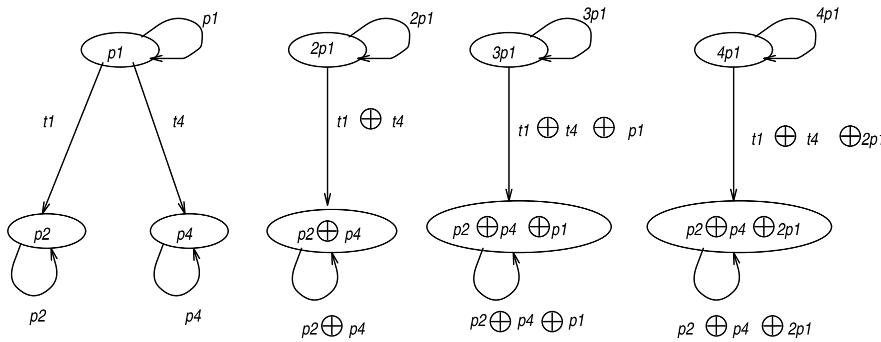


Figure 12: Reflexive parallel P/T-Graph of the reader-writer system

For instance, in the parallel P/T-Graph corresponding to the Figure 10, the firing step $t_4 \oplus t_1$ can fire from $2p_1$ to $p_2 \oplus p_4$ but not from the marking $3p_1$, since $3p_1$ is an isolated vertex in the parallel P/T-Graph. Therefore, in order to derive the marking graph from the P/T-Graph, it is first necessary to extend our notion of P/T-Monoid in the following way. For each place $x \in P$ we introduce a **reflexive transition** with the same name, which has $x$ itself as both pre- and post-condition (essentially an identity loop). As an example, consider a subnet of our reader-writer system in Figure 11. A subnet of the reflexive monoid with the reflexive transitions can be seen in Figure 12. Observe that we now have the "missed" transition from $3p_1$, namely $3p_1[t_1 \oplus t_4 \oplus p_1\rangle p_2 \oplus p_4 \oplus p_1$.

Note that the firing of a transition under a specific marking which is bigger than the pre-condition of the respective transition (containing therefore the so-called "idle" tokens) corresponds, in the reflexive P/T-Monoid, to the parallel composition of the firing transition with so many idle tokens as the ones which are not "consumed" by the transition. As an example, consider the above mentioned transition $t_1 \oplus t_4$. Its pre-condition is $pre^{\oplus}(t_1 \oplus t_4) = pre(t_1) \oplus pre(t_4) = 2p_1$. Now taking into account the vertex (marking) $4p_1$ we have that $4p_1 - 2p_1 = 2p_1$, i.e., we have two idle tokens $p_1$. This means that the actual parallel transition from the pre-condition $4p_1$ should be $t_1 \oplus t_4 \oplus 2p_1$. Its respective post-condition is $post^{\oplus}(t_1 \oplus t_4 \oplus 2p_1) = post(t_1) \oplus post(t_4) \oplus post(2p_1) = p_2 \oplus p_4 \oplus 2p_1$, where $post(2p_1) = 2p_1$ holds because every vertex in the reflexive P/T-Monoid should also be equipped with an identity loop as discussed previously.

This discussion motivates us to introduce the following

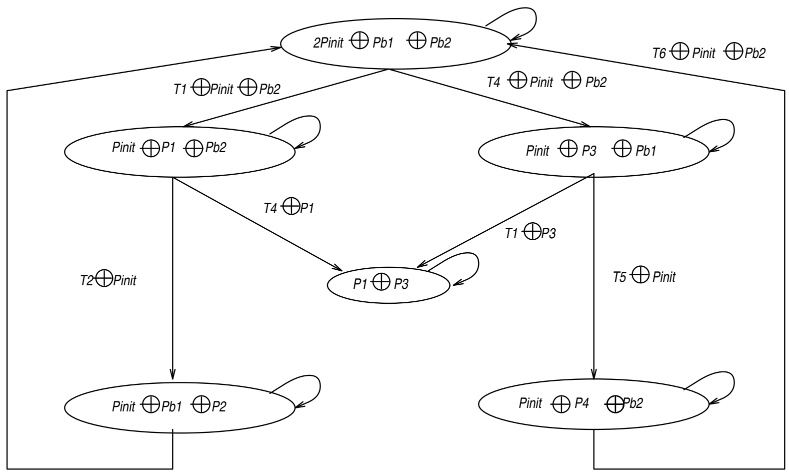


Figure 13: Reflexive P/T-Monoid corresponding to the sequential reachability graph of the deadlock system of Figure 3

**Definition 4.2.1** *A reflexive P/T-Monoid* $\mathcal{R} =_{def} (P, P^{\oplus}, \mathcal{T} =_{def} (U, \epsilon, +), pre, post : (U, \epsilon, +) \to P^{\oplus})$ *is a P/T-Monoid together with a monoid homomorphism* $idle : P^{\oplus} \to (U, \epsilon, +)$*, which associates an identity transition (edge) to each vertex from* $P^{\oplus}$ *such that* $pre \circ idle = post \circ idle = id_{P^{\oplus}}$*.*

**Example 4.2.2** A reflexive P/T-Monoid corresponds in some sense to the marking graph of definition 2.13, with the crucial difference that in the reflexive P/T-Monoid all possible parallel firings are also present. On the other hand, since marking graphs are essentially infinite, we take, as an example, the finite, sequential reachability graph of the deadlock system of Figure 2. The corresponding reflexive P/T-Monoid is depicted in Figure 13. This graph is, up to "loops", isomorphic to the sequential reachability graph.

**Definition 4.2.3** *Let* $\mathcal{R}_i =_{def} (P_i, P_i^{\oplus}, (U_i, \epsilon, +), pre_i, post_i : (U_i, \epsilon, +) \to P_i^{\oplus}, idle_i : P_i^{\oplus})$*,* $i = 1, 2$ *be reflexive P/T-Monoids. A reflexive* $P/T-Monoid$ *morphism is a pair of functions* $f =_{def} \langle f_P, f_{\mathcal{T}} \rangle$*, where* $f_P : P_1 \to P_2$ *and* $f_{\mathcal{T}} : (U_1, \epsilon, +) \to (U_2, \epsilon, +)$ *is a monoid homomorphism such the following diagram should commute componentwise, i.e.,* $f^{\oplus} \circ pre_1^{\oplus} = pre_2^{\oplus} \circ f_{\mathcal{T}}$ *(the same for* $post^{\oplus}$*) and* $idle_2 \circ f_P^{\oplus} = f_{\mathcal{T}} \circ idle_1$*:*

$$
\begin{tikzcd}
(U_1, \epsilon, +) \arrow[r, shift left, "pre_1^{\oplus}"] \arrow[r, shift right, "post_1^{\oplus}"'] \arrow[d, "f_{\mathcal{T}}"'] & P_1^{\oplus} \arrow[l, bend right, "idle_1"'] \arrow[d, "f_P^{\oplus}"] \\
(U_2, \epsilon, +) \arrow[r, shift left, "pre_2^{\oplus}"] \arrow[r, shift right, "post_2^{\oplus}"'] & P_2^{\oplus} \arrow[l, bend left, "idle_2"]
\end{tikzcd}
$$

**Remark 4.2.4** In general, reflexive P/T-Monoids are nothing else than reflexive graphs. A reflexive graph $(G, id_G : V \to E)$ where $G =_{def} (V, E, s, t : E \to V)$ is a graph such that for each $v \in V$ there exists a corresponding identity edge (loop) $id_G(v)$ such that $s(id_G(v)) = t(id_G(v)) = v$. Now, reflexive graph morphisms are graph morphisms $f =_{def} \langle f_V, f_E \rangle : G_1 \to G_2$ which should additionally preserve identity loops, i.e., such that the equation $f_E \circ id_{G_1}(v) = id_{G_2} \circ f_V(v)$ should hold for all $v \in V_1$.

**Proposition 4.2.5** *The class of all reflexive P/T-Monoids together with the set of all reflexive P/T-Monoid morphisms constitute a category, called* **RM-Graph**

**Proof:** Given reflexive P/T-Monoid morphisms $f : \mathcal{N}_1^{\oplus} \to \mathcal{N}_2^{\oplus}, g : \mathcal{N}_2^{\oplus} \to \mathcal{N}_3^{\oplus}$, we have to show that the following diagram commutes as well as the preservation of reflexive transitions.

$$
\begin{tikzcd}
(U_1, \epsilon, +) \arrow[r, shift left, "pre_1^{\oplus}"] \arrow[r, shift right, "post_1^{\oplus}"'] \arrow[d, "f_{\mathcal{T}}"'] & P_1^{\oplus} \arrow[l, bend right=40, "idle_1"'] \arrow[d, "f_P^{\oplus}"] \\
(U_2, \epsilon, +) \arrow[r, shift left, "pre_2^{\oplus}"] \arrow[r, shift right, "post_2^{\oplus}"'] \arrow[d, "g_{\mathcal{T}}"'] & P_2^{\oplus} \arrow[l, bend left=40, "idle_2"] \arrow[d, "g_P^{\oplus}"] \\
(T_3, \oplus, \epsilon) \arrow[r, shift left, "pre_3^{\oplus}"] \arrow[r, shift right, "post_3^{\oplus}"'] & P_1^{\oplus} \arrow[l, bend left=40, "idle_3"]
\end{tikzcd}
$$

Now observe that the bigger square commute (componentwise) by assumption, since $f$ and $g$ are reflexive P/T-Monoid morphisms. Preservation of identity transitions hold by the following trivial calculation:

$$
\begin{array}{rcll}
idle_3 \circ g_P^{\oplus} \circ f_P^{\oplus} & = & f_{\mathcal{T}} \circ idle_2 \circ f_P^{\oplus} & (g \text{ is a ref. P/T-Monoid morph.}) \\
& = & g_{\mathcal{T}} \circ f_{\mathcal{T}} \circ idle_1 & (f \text{ is a ref. P/T-Monoid morph.})
\end{array}
$$

This shows that $g \circ f =_{def} \langle g_P \circ f_P, g_{\mathcal{T}} \circ f_{\mathcal{T}} \rangle$ is a reflexive P/T-Monoid morphism. The associativity law holds because for all $f : \mathcal{N}_1^{\oplus} \to \mathcal{N}_2^{\oplus}, g : \mathcal{N}_2^{\oplus} \to \mathcal{N}_3^{\oplus}, h : \mathcal{N}_3^{\oplus} \to \mathcal{N}_4^{\oplus}$ we have that $h \circ (g \circ f) = (h \circ g) \circ f$ since the components of reflexive P/T-Monoid morphisms are functions in **Set**, respectively, monoid homomorphisms in **CMon**. The identity law holds also by a similar argument. □

**Definition 4.2.6 Marking Graph** *Given a P/T-Net* $\mathcal{N} =_{def} (P, P^{\oplus}, T, pre, post : T \to P^{\oplus})$*, the corresponding marking graph* $M[N]$ *is given by* $M[N] =_{def} (P, P^{\oplus}, T^{\oplus} + P^{\oplus}, [pre^{\oplus}, in_P^{\oplus}], [post^{\oplus}, in_P^{\oplus}] : T^{\oplus} + P^{\oplus} \to P^{\oplus}, idle : P^{\oplus} \to T^{\oplus} + P^{\oplus})$*, where* $in_P^{\oplus}$ *is the inclusion of* $P^{\oplus}$ *into* $: T^{\oplus} + P^{\oplus}$*.*

**Example 4.2.7** Figure 14 shows a subset of the marking graph corresponding to the example of the deadlock system. This graph is also a subgraph of the marking graph in the sense of 2.13. The complete (infinite) marking graph contains additionally many similar "parallel reachability graphs", corresponding to others initial markings. Especially, this graph corresponds to the "initial" marking $M =_{def} 3Pinit \oplus 2Pb1$.

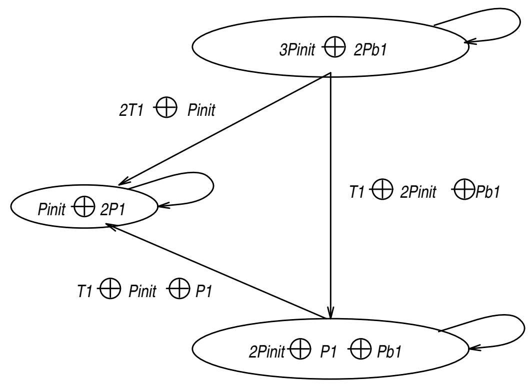


Figure 14: Subgraph of the marking graph of the deadlock P/T-net as a reflexive P/T-Monoid

The construction of the marking graph can be described functorially, more specifically by a free functor, which for each P/T-Graph associates its corresponding marking graph. The precise mathematical formulation of this construction is nothing but the task of our next

**Theorem 4.2.8** *There is a free functor $F$ :***P/T-Net→RM-Graph***, which delivers a marking graph $M[N]$ in* **RM-Graph** *for each Petri Net $N$ in* **P/T-Net***. Moreover, this free functor is left adjoint with respect to the forgetful functor $V$ :***RM-Graph→ P/T-Net***, which forgets reflexive transitions and parallel composition of transitions.*

**Proof:** We must show that for each P/T-Net $\mathcal{N} =_{def} (P, P^{\oplus}, T, pre, post : T \to P^{\oplus})$, for each reflexive P/T-Monoid $\mathcal{M} =_{def} (R, R^{\oplus}, (U, \epsilon, +), pre_M, post_M : (U, \epsilon, +) \to R^{\oplus}, idle_M : R^{\oplus} \to (U, \epsilon, +))$ and for each **P/T-Net** morphism $f =_{def} \langle f_P, f_T \rangle : N \to V(\mathcal{M})$, where $V(\mathcal{M}) =_{def} (R, R^{\oplus}, U, pre_M, post_M : U \to R^{\oplus})$, $f_P : P \to R$, $f_T : T \to U$, there is a unique $f^{\sharp} : F(N) \to M$ such that the following diagram commutes:

$$\begin{tikzcd}
N \arrow[r, "f"] \arrow[d, "\eta_N"'] & V(\mathcal{M}) \\
V(F(N)) \arrow[ur, "V(f^{\sharp})"'] &
\end{tikzcd}$$

We define $F(N)$, according to 4.2.6, as $F(N) =_{def} (P, P^{\oplus}, T^{\oplus}+P^{\oplus}, [pre^{\oplus}, in_P^{\oplus}], [post^{\oplus}, in_P^{\oplus}] : T^{\oplus}+P^{\oplus} \to P^{\oplus}, idle : P^{\oplus} \to T^{\oplus}+P^{\oplus})$ and $V(F(N)) =_{def} (P, P^{\oplus}, T^{\oplus}+P^{\oplus}, [pre^{\oplus}, in_P^{\oplus}], [post^{\oplus}, in_P^{\oplus}] : V(T^{\oplus} + P^{\oplus}) \to P^{\oplus})$.

We first define the universal morphism $\eta_N$ as $\eta_N =_{def} \langle id_P, in_T \rangle$ such that the following diagram trivially commutes, i.e., such that $\eta_N$ becomes a **P/T-Net** morphism.

$$\begin{tikzcd}
T \arrow[r, shift left, "pre"] \arrow[r, shift right, "post"'] \arrow[d, "in_T"'] & P^{\oplus} \arrow[d, "id_P^{\oplus}"] \\
V(T^{\oplus} + P^{\oplus}) \arrow[r, shift left, "{[pre^{\oplus}, in_P^{\oplus}]}"] \arrow[r, shift right, "{[post^{\oplus}, in_P^{\oplus}]}"'] & P^{\oplus}
\end{tikzcd}$$

To existence of $f^\sharp$, note that, by assumption, we have a P/T-Net morphism $f =_{def} \langle f_P, f_T \rangle$ such that the following diagram commutes (where $U = V((U, \epsilon, +))$).

$$\begin{tikzcd}
T \arrow[r, shift left, "pre"] \arrow[r, shift right, "post"'] \arrow[d, "f_T"'] & P^{\oplus} \arrow[d, "f_P^{\oplus}"] \\
U \arrow[r, shift left, "pre_M"] \arrow[r, shift right, "post_M"'] & R^{\oplus}
\end{tikzcd}$$

Now consider the following (coproduct) diagram in **Set**:

$$\begin{tikzcd}
P \arrow[r, "in_P"'] & P \uplus T & T \arrow[l, "in_T"]
\end{tikzcd}$$

Since $F$ preserve colimits (according to A.6.5), we also obtain the following coproduct diagram in **CMon**:

$$\begin{tikzcd}
P^{\oplus} \arrow[r, "in_P^{\oplus}"'] & P^{\oplus} + T^{\oplus} & T^{\oplus} \arrow[l, "in_T^{\oplus}"]
\end{tikzcd}$$

This means that, given **CMon** morphisms $idle_M \circ f_P^{\oplus} : P^{\oplus} \to R^{\oplus} \to (U, \epsilon, +)$, $f_T^{\oplus} : T^{\oplus} \to (U, \epsilon, +)$, there is a unique $h =_{def} [idle_M \circ f_P^{\oplus}, f_T^{\oplus}]$, by the couniversal property of $P^{\oplus} + T^{\oplus}$, such that the following diagram commutes:

$$\begin{tikzcd}
 & (U, \epsilon, +) & \\
P^{\oplus} \arrow[ur, "idle_M \circ f_P^{\oplus}"] \arrow[r, "in_P^{\oplus}"'] & P^{\oplus} + T^{\oplus} \arrow[u, "h"] & T^{\oplus} \arrow[l, "in_T^{\oplus}"] \arrow[ul, "f_T^{\oplus}"']
\end{tikzcd}$$

Now, we define $f^\sharp =_{def} \langle f_P, [idle_M \circ f_P^{\oplus}, f_T^{\oplus}] \rangle$, and to verify that this is well-defined, the following diagram has to be shown commutative, which we will leave as an exercise for the reader:

$$\begin{tikzcd}
T^{\oplus} + P^{\oplus} \arrow[r, shift left, "{[pre^{\oplus}, id_P^{\oplus}]}"] \arrow[r, shift right, "{[post^{\oplus}, id_P^{\oplus}]}"'] \arrow[d, "{[idle_M \circ f_P^{\oplus}, f_T^{\oplus}]}"'] & P^{\oplus} \arrow[d, "f_P^{\oplus}"] \arrow[l, bend right=50, "idle"'] \\
(U, \epsilon, +) \arrow[r, shift left, "pre_M"] \arrow[r, shift right, "post_M"'] & R^{\oplus} \arrow[l, bend left=50, "idle_M"]
\end{tikzcd}$$

To see that $V(f^\sharp) \circ \eta_N = f$, note first that $V(f^\sharp) = f^\sharp$, since the base sets of $F(N)$ are kept after the application of the forgetful functor $V$. Moreover, we have that

$$
\begin{aligned}
V(f^\sharp)\circ\eta_N &= f^\sharp\circ\eta_N \\
&\qquad\text{(by the above remark)} \\
&= \langle f_P, [idle_M\circ f_P^\oplus, f_T^\oplus]\rangle\circ\langle id_P, in_T\rangle \\
&\qquad\text{(by definition of } f^\sharp \text{ and } \eta_N) \\
&= \langle f_P, f_T\rangle \\
&\text{(identity and the definition of the induced coproduct morphism)} \\
&= f \\
&\qquad\text{(by definition)}
\end{aligned}
$$

Please, note that the second equality above holds, for all $t \in T$, by the following trivial calculation:

$$
\begin{aligned}
[idle_M\circ f_P^\oplus, f_T^\oplus]\circ in_T(t) &= [idle_M\circ f_P^\oplus, f_T^\oplus](t) \\
&\text{(by definition of inclusion)} \\
&= f_T^\oplus(t) \\
&\text{(by definition and remark below)} \\
&= f_T(t) \\
&\text{(by definition of a free extension)}
\end{aligned}
$$

To see that $f^\sharp$ is the unique morphism that satisfies such commutativity, consider another $g =_{def} \langle g_P, g_T\rangle, g : P \to R, g_T : P^\oplus + T^\oplus \to (U, \epsilon, +)$ which fulfills the same requirement, i.e., $V(g)\circ\eta_N = f$. Since we have $V(g) = g$ we have that that $\langle g_P, g_T\rangle\circ\langle id_P, in_T\rangle = \langle f_P, [idle_M\circ f_P^\oplus, f_T^\oplus]\rangle$ and then, by composition and the identity law, $g_P = f_P, g_T = g_T\circ in_T = [idle_M\circ f_P^\oplus, f_T^\oplus]$. This, after all, means that $g = f^\sharp$, as required. □

## 5 Bibliographic notes

The starting point for Petri net theory dates from Dr. Petri's Ph.D. dissertation [13]. Good introductions on Petri nets in general can be found in [11, 12, 14], where the first is a tutorial and the other two are textbooks. Especial thanks for Claudia Ermel's highly readable Studienarbeit [5], from which I have "borrowed" the examples and the basis for this presentation. Other topics closely related to the discussion in this text are the notions of Algebraic High-Level Nets (see [10, 16, 15]) where tokens are "structured entities" (instead of black dots), more precisely, terms from an algebraic specification. The milestone paper which set the basis for an algebraic approach to the study of Petri Nets is [8]. A very recent work connecting category theory and semantic foundations for a unifying view of several classes of Petri Nets is Julia Padberg's PhD thesis [9].

**Acknowledgments:** This work was developed within a German-Brazilian cooperation on Information Technology, in the context of GRAPHIT project, and partially supported by a CNPq-grant 200529/94-3 for Álfio Martini. This is part of a work which was initiated by Prof. Dr. Harmut Ehrig's Kloosterman Lecture "On the role of Category Theory in

Computer Science" in Leiden, Holland, September, 1993, as well as the encouragement of Prof. Dr. Daltro Nunes concerning categorical methods in computer science. A very special "thank you" goes to Uwe Wolter, who carefully read this report, found numerous significant errors, and made very useful suggestions on how to correct them.

# A Categorical Background

In this appendix we summarize some basic notions from category theory which are used in this report as well as a set of elementary results (propositions and theorems) concerning categorical constructions that are referenced in the proofs. The complete proofs of the following elementary results, may be found, for instance, in [7]. Detailed introductions to this subject may be found in [1], [2], and [6].

## A.1 Categories and diagrams

**Definition A.1.1** *A category* $\mathbf{C}$ *comprises*

1. *a collection* $Ob(\mathbf{C})$ *of* **objects***;*
2. *a collection* $Mor(\mathbf{C})$ *of* **morphisms***;*
3. *two operations* $dom, cod : Mor(\mathbf{C}) \rightarrow Ob(\mathbf{C})$ *assigning to each morphism* $f$ *two objects, called respectively* **domain** *and* **codomain** *of* $f$*;*
4. *a composition operator* $\circ : Mor(\mathbf{C}) \times Mor(\mathbf{C}) \rightarrow Mor(\mathbf{C})$ *assigning to each pair of morphisms* $\langle f, g \rangle$ *with* $dom(g) = cod(f)$ *a composite morphism* $g \circ f : dom(f) \rightarrow cod(g)$*, such that the following* **associative law** *holds:*

   *For any morphisms* $f, g, h$ *in* $Mor(\mathbf{C})$ *such that* $cod(f) = dom(g) \wedge cod(g) = dom(h)$*:*

$$h \circ (f \circ g) = (h \circ g) \circ h$$

5. *for each object* $A$*, an identity morphism* $id_A : A \rightarrow A$*, such that the following* **identity law** *holds:*

   *For any morphism* $f$ *such that* $dom(f) = A \wedge cod(f) = B$*:*

$$id_B \circ f = f \wedge f \circ id_A = f$$

**Remarks A.1.2**

1. Category theory is based on composition as a fundamental operation, in much the same way that classical set theory is based on the "element of" or membership relation.
2. Categories will be denoted by uppercase boldface letters $\mathbf{A}, \mathbf{B}, \mathbf{C}, \ldots$ from the beginning of the alphabet.
3. We use letters $A, B, \ldots, Y, Z$ from the alphabet (with subscripts when appropriate) to denote objects and lowercase letters $a, b, c, \ldots, f, g, h, \ldots, y, z$ (occasionally with subscripts) to denote morphisms in any category.
4. If $dom(f) = A \wedge cod(f) = B$ we write $f : A \rightarrow B$ to denote that $f$ is a morphism from $A$ to $B$.

| *CATEGORY* | *OBJECTS* | *MORPHISMS* |
|---|---|---|
| **Set** | sets | total functions |
| **FinSet** | finite sets | total functions |
| **Pfn** | sets | partial functions |
| **Rel** | sets | binary relations |
| **Mon** | monoids | monoid homomorphisms |
| **Poset** | posets | monotonic functions |
| **Grp** | groups | group homomorphisms |
| $\Sigma$-**Alg** | $\Sigma$-Alg | $\Sigma$-homomorphisms |
| **Cat**$(SPEC)$ | $SPEC$-algebras | $SPEC$-homomorphisms |
| **Aut** | finite automata | automata homomorphisms |
| **Graph** | directed graphs | graph morphisms |

Table 1: Examples of categories

5. The collection of all morphisms with domain $A$ and codomain $B$ will be written as $\mathbf{C}(A, B)$ or as $Hom_{\mathbf{C}}(A, B)$.
6. Given a category $\mathbf{C}$, we may write "$\mathbf{C}$-object $A$" (resp. "$\mathbf{C}$-morphism $f : A \to B$") or "$A \in Ob(\mathbf{C})$" (resp. "$f \in \mathbf{C}(A, B)$") to denote that $A$ is a object of $\mathbf{C}$ (resp. $f$ is a morphism from $A$ to $B$ in $\mathbf{C}$).

**Example A.1.3** Table 1 lists some categories by specifying their objects and morphisms.

**Definition A.1.4** *A category* $\mathbf{D}$ *is a* **subcategory** *of a category* $\mathbf{C}$ *if:*

1. $Ob(\mathbf{D}) \subseteq Ob(\mathbf{C})$*;*
2. *for all* $A,\ B \in Ob(\mathbf{D})$*,* $\mathbf{D}(A, B) \subseteq \mathbf{C}(A, B)$*;*
3. *compositions and identities coincide with those of* $\mathbf{C}$*.*

**Definition A.1.5** *A subcategory* $\mathbf{D}$ *of* $\mathbf{C}$ *is said to be* **full** *if for all* $A,\ B \in Ob(\mathbf{D})$*,* $\mathbf{D}(A, B) = \mathbf{C}(A, B)$*.*

**Example A.1.6** For any category $\mathbf{C}$, the empty category $\mathbf{0}$ and $\mathbf{C}$ itself are full subcategories of $\mathbf{C}$.

**Example A.1.7** The category **FinSet** is a full subcategory of **Set**

**Example A.1.8** The category **Set** is a subcategory of **Pfn**. However, it is not a full subcategory, since between any two sets there are *much* more partial functions than total functions. Besides, **Pfn** is a subcategory of **Rel**. It is, however, not a full subcategory.

**Definition A.1.9** *A* **diagram** *in a category* $\mathbf{C}$ *is a collection of vertices and directed edges, consistently labeled with objects and morphisms of* $\mathbf{C}$*, where "consistently" means that if an edge in the diagram is labeled with a morphism* $f$*, and* $f$ *has domain* $A$ *and codomain* $B$*, then the endpoints of this edge must be labeled with* $A$ *and* $B$*.*

**Definition A.1.10** *A diagram in a category* $\mathbf{C}$ *is said to* **commute** *(or is said to be* **commutative***) if, for each pair of vertices* $v_1$ *and* $v_2$ *labeled by* $X$ *and* $Y$*, all paths in the diagram from* $X$ *to* $Y$ *are equal, in the sense that each path in the diagram determines a composite morphism and these composites are equal in* $\mathbf{C}$*.*

As an example, saying that the diagram below commutes is exactly the same as saying that $f \circ g' = g \circ f'$.

$$\begin{tikzcd} X \arrow[r, "f'"] \arrow[d, "g'"'] & Z \arrow[d, "g"] \\ W \arrow[r, "f"'] & Y \end{tikzcd}$$

## A.2 Special kinds of morphisms

**Definition A.2.1** *Let* $\mathbf{C}$ *be a category. A morphism* $f \in \mathbf{C}(B, C)$ *is a* **monomorphism** *(or is* **monic***) if for any pair of morphisms* $g \in \mathbf{C}(A, B)$ *and* $h \in \mathbf{C}(A, B)$*, the equality* $f \circ g = f \circ h$ *implies* $g = h$ *(i.e.,* $f$ *is left-cancelable with respect to composition).*

**Example A.2.2** In **Set** a function is a monomorphism iff it is injective.

**Definition A.2.3** *Let* $\mathbf{C}$ *be a category. A morphism* $f \in \mathbf{C}(A, B)$ *is an epimorphism (or* **comonomorphism** *or only* **epic***) if for any pair of morphisms* $g \in \mathbf{C}(B, C)$ *and* $h \in \mathbf{C}(B, C)$*, the equality* $g \circ f = h \circ f$ *implies* $g = h$ *(i.e.,* $f$ *is right-cancelable with respect to composition).*

**Example A.2.4** In **Set** a function is an epimorphism iff it is surjective.

## A.3 Universal constructions

**Definition A.3.1** *Let* $\mathbf{C}$ *be a category. An object* $A$ *is a* **initial object** *in* $\mathbf{C}$ *if for any* $\mathbf{C}$*-object* $A$ *there exists a unique morphism* $f \in \mathbf{C}(A, B)$

**Example A.3.2** In **Set** and **Pfn** the initial object is the emptyset.

**Definition A.3.3** A **coproduct** of two objects $A$ and $B$ in a category $\mathbf{C}$ is a $\mathbf{C}$-object $A+B$ together with two **injection morphisms** $in_A \in \mathbf{C}(A, A+B)$ and $in_B \in \mathbf{C}(B, A+B)$, such that for any $\mathbf{C}$-object $C$ and any pair of morphisms $f \in \mathbf{C}(A, C)$ and $g \in \mathbf{C}(B, C)$, there exists exactly one morphism $h \in \mathbf{C}(A+B, C)$, such that $h \circ in_A = f$ and $h \circ in_B = g$ (see next diagram).

$$\begin{tikzcd} & C & \\ A \arrow[ur, "f"] \arrow[r, "in_A"'] & A+B \arrow[u, "h"] & B \arrow[ul, "g"'] \arrow[l, "in_B"] \end{tikzcd}$$

**Example A.3.4** In **Set** the coproduct of two sets $A$ and $B$ is given by the disjoint union $A \uplus B$.

**Definition A.3.5** *A morphism $c \in \mathbf{C}(A, X)$ is a* **coequalizer** *of a pair of morphisms $f \in \mathbf{C}(B, A)$ and $g \in \mathbf{C}(B, A)$ if*

1. *$c \circ f = c \circ g$;*
2. *whenever $c' \in \mathbf{C}(A, X')$ satisfies $c' \circ f = c' \circ g$, then there exists a unique morphism $k \in \mathbf{C}(X, X')$ such that $k \circ c = c'$, as shown in the next diagram.*

$$\begin{tikzcd} B \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & A \arrow[r, "c"] \arrow[dr, "c'"'] & X \arrow[d, "k"] \\ & & X' \end{tikzcd}$$

**Example A.3.6** In **Set** coequalizers can be constructed by using equivalence relations. Given two functions $f, g : X \to A$, the intuitive idea is to identify $f(x)$ with $g(x)$ for all $x \in X$. First we construct the relation $R = \{\langle f(x), g(x)\rangle\}$ for all $x \in X$. Next we define $R'$ as being the least equivalence relation which contains $R'$ (i.e., the reflexive, symmetric and transitive closure of $R$). Then we define the coequalizer $nat : A \to A/R'$ by $nat(a) = [a]$ (also called "natural" mapping).

**Proposition A.3.7** *Every coequalizer is an epimorphism.*

**Definition A.3.8** *A* **pushout** *of a pair of morphisms $f \in \mathbf{C}(C, A)$ and $g \in \mathbf{C}(C, B)$ is a $\mathbf{C}$-object $P$ and a pair of morphisms $g' \in \mathbf{C}(A, P)$ and $f' \in \mathbf{C}(B, P)$, such that*

1. *$g' \circ f = f' \circ g$;*
2. *for any other $\mathbf{C}$-object $X$ and pair of morphisms $i \in \mathbf{C}(A, X)$, $j \in \mathbf{C}(B, X)$ with $i \circ f = j \circ g$, there exists only one morphism $k : P \to X$ such that $i = k \circ g'$ and $j = k \circ f'$, as shown by the following diagram:*

$$\begin{tikzcd} C \arrow[r, "g"] \arrow[d, "f"'] & B \arrow[d, "f'"] \arrow[ddr, "j"] & \\ A \arrow[r, "g'"] \arrow[drr, "i"'] & P \arrow[dr, dashed, "k"] & \\ & & X \end{tikzcd}$$

## A.4 Functors and natural transformations

**Definition A.4.1** *Let* $\mathbf{C}$ *and* $\mathbf{D}$ *be categories. A (covariant)* **functor** $F : C \to D$ *is a pair of mappings,* $F_{Ob} : Ob(\mathbf{C}) \to Ob(\mathbf{D})$*,* $F_{Mor} : Mor(\mathbf{C}) \to Mor(\mathbf{D})$ *for which*

1. *If* $f : A \to B$ *in* $\mathbf{C}$*, then* $F_{Mor}(f) : F_{Ob}(A) \to F_{Ob}(B)$ *in* $\mathbf{D}$*.*
2. $F_{Mor}(id_A) = id_{F_{Ob}(A)}$*;*
3. $F_{Mor}(g \circ f) = F_{Mor}(g) \circ F_{Mor}(f)$*.*

**Remark A.4.2** It is usual practice to omit the subscripts "Ob" and "Mor" as it always clear from context whether the functor is meant to operate on objects or on morphisms.

**Example A.4.3** For every category $\mathbf{C}$, we have the identity functor $I_{\mathbf{C}}$ which takes each $\mathbf{C}$-object and every $\mathbf{C}$-morphism to itself. More precisely we have that $I_{\mathbf{C}}(g \circ f) = g \circ f = I_{\mathbf{C}}(g) \circ I_{\mathbf{C}}(f)$, and $I_{\mathbf{C}}(id_A) = id_A = id_{I_{\mathbf{C}}(A)}$.

**Example A.4.4 Forgetful functors**. Forgetting some of the structure in a category of structures and structure-preserving functions gives a functor called **underlying functor** or **forgetful functor**. The functor $V : \mathbf{Mon} \to \mathbf{Set}$ which sends each monoid $(M, \cdot, e)$ to the set $M$ and each monoid homomorphism $h : (M, \cdot, e) \to (M', \cdot'.e')$ to the corresponding function $h : M \to M'$ on the underlying sets is a forgetful functor. Note that we have $V(id_{(M,\cdot,e)}) = id_M = id_{V(M,\cdot,e)}$ and $V(g \circ f) = g \circ f = V(g) \circ V(f)$ for every $f : (M, \cdot, e) \to (M', \cdot', e')$, $g : (M', \cdot', e') :\to (M'', \cdot'', e'')$ and $id_{(M,\cdot,e)} : (M, \cdot, e) \to (M, \cdot, e)$.

**Definition A.4.5** *Let* $\mathbf{C}$ *and* $\mathbf{D}$ *be categories and* $F$ *and* $G$ *be functors from* $\mathbf{C}$ *to* $\mathbf{D}$*. A* **natural transformation** $\eta$ *from* $F$ *to* $G$*, written* $\eta : F \dot{\to} G$*, is a function that assigns to every* $\mathbf{C}$*-object* $A$ *a* $\mathbf{D}$*-morphism* $\eta_A : F(A) \to G(A)$ *such that for any* $\mathbf{C}$*-arrow* $f : A \to B$ *the diagram on the right commutes in* $\mathbf{D}$*:*

$$
\begin{array}{ccc}
A & \quad & \begin{tikzcd} F(A) \arrow[r, "\eta_A"] \arrow[d, "F(f)"'] & G(A) \arrow[d, "G(f)"] \\ F(B) \arrow[r, "\eta_B"] & G(B) \end{tikzcd} \\
\Big\downarrow f & & \\
B & &
\end{array}
$$

**Definition A.4.6** *Let* $F, G$ *be functors from* $\mathbf{C}$ *to* $\mathbf{D}$*. If each component* $\eta_A$ *of* $\eta : F \dot{\to} G$ *is an isomorphism in* $\mathbf{D}$*, we call* $\eta$ *a* **natural isomorphism***.*

**Proposition A.4.7** *Suppose* $F : \mathbf{C} \to \mathbf{D}$ *and* $G : \mathbf{C} \to \mathbf{D}$ *are functors and* $\eta : F \dot{\to} G$ *is natural isomorphism. Then there is a unique natural transformation* $\eta^{-1} : G \dot{\to} F$ *such that the composites* $\eta \circ \eta^{-1} = \iota_G$ *and* $\eta^{-1} \circ \eta = \iota_F$*.*

## A.5 General colimit constructions

**Definition A.5.1** *A* **cocone** *for a diagram* $\mathbf{D}$ *in a category* $\mathbf{C}$ *is a* $\mathbf{C}$*-object* $X$ *and a collection of* $\mathbf{C}$*-morphisms* $f_i : D_i \to X$ *(one for each* $D_i$ *in* $\mathbf{D}$*) such that the following diagram commutes:*

$$\begin{tikzcd} & X & \\ D_i \arrow[ur, "f_i"] \arrow[rr, "g"'] & & D_j \arrow[ul, "f_j"'] \end{tikzcd}$$

**Definition A.5.2** *A* **colimit** *or* **inverse limit** *for* $\mathbf{D}$ *is then a cocone* $\{f_i : D_i \to X\}$ *with the couniversal property that for any other cocone* $\{f_i' : D_i \to X'\}$ *there is a unique morphism* $k : X \to X'$ *such that the diagram*

$$\begin{tikzcd} X' & & X \arrow[ll, "k"'] \\ & D_I \arrow[ul, "f_i'"] \arrow[ur, "f_i"'] & \end{tikzcd}$$

*commutes for every* $D_i$ *in* $\mathbf{D}$*.*

**Proposition A.5.3** *Colimits are unique up to isomorphism.*

**Example A.5.4** Initial objects, coproducts, coequalizers and pushouts are all special cases of colimit constructions.

**Definition A.5.5** *A* **finite diagram** *is one that has a finite number of vertices and a finite number of edges between them.*

**Definition A.5.6** *A* **small diagram** *is a diagram whose collection of vertices and edges are really sets and not proper classes.*

**Definition A.5.7** *A category is said to*

1. **have (finite) coproducts** *provided that for each (finite) set-indexed family of objects there exists a coproduct;*

2. **have coequalizers** *provided that for each pair of morphisms (with same domain and codomain) there exists an coequalizer;*

3. **have pushouts** *provided that for each pair of morphisms with the same domain there exists an pullback.*

**Definition A.5.8** *A category* $\mathbf{C}$ *is said to be*

1. **finitely cocomplete** *if for each finite diagram in* $\mathbf{C}$ *there exists a colimit;*
2. **cocomplete** *if for each small diagram in* $\mathbf{A}$ *there exists a colimit.*

**Theorem A.5.9** *For each category* $\mathbf{C}$*, the following conditions are equivalent:*

1. $\mathbf{C}$ *is finitely cocomplete.*
2. $\mathbf{C}$ *has finite coproducts and coequalizers.*
3. $\mathbf{C}$ *has pushouts and an initial object.*

**Theorem A.5.10** *For each category* $\mathbf{C}$*, the following conditions are equivalent:*

1. $\mathbf{C}$ *is cocomplete.*
2. $\mathbf{C}$ *has coproducts and coequalizers.*
3. $\mathbf{C}$ *has coproducts and pushouts.*

## A.6 Free constructions and adjoints

**Definition A.6.1** *Let* $G : \mathbf{D} \to \mathbf{C}$ *be a functor. We call an object* $D$ *in* $\mathbf{D}$ *a* **free construction over** $A$ **with respect to** $G$*, if there is a morphism* $\eta_A : A \to G(D)$ *in* $\mathbf{A}$*, called* **universal morphism***, such that for any morphism* $f : A \to G(B)$ $(A \in \mathbf{C}, B \in \mathbf{D})$ *there is a unique morphism* $f^\sharp : D \to B$ *in* $\mathbf{D}$ *such that* $G(f^\sharp) \circ \eta_A = f$*. In this case we say that the following diagram commutes:*

$$\begin{tikzcd}
A \arrow[r, "f"] \arrow[d, "\eta_A"'] & G(B) \\
G(D) \arrow[ur, dotted, "G(f^\sharp)"'] &
\end{tikzcd}$$

**Example A.6.2** For each set $P, \mathcal{P} =_{def} (P^*, \lambda, \cdot)$ is called the free monoid generated by $P$, where $P^*$ is the set of all words over $P$, $\lambda$ is the empty word, and $\cdot$ the concatenation operation.

**Theorem A.6.3** *Consider the functor* $G : \mathbf{D} \to \mathbf{C}$*. Suppose that for each* $A$ *of* $\mathbf{C}$*, a free object* $D \in \mathbf{D}$ *with universal morphism* $\eta_A$ *exists. Then the free construction* $D$ *can be extended to morphisms in* $\mathbf{C}$ *such that for each* $h : A \to B$ *in* $\mathbf{C}$*,* $(\eta_B \circ h)^\sharp : D \to D'$ *is uniquely defined by commutativity of the following diagram:*

$$
\begin{tikzcd}
I_{\mathbf{C}}(A) \arrow[r, "I_{\mathbf{C}}(h)"] \arrow[d, "\eta_A"'] & I_{\mathbf{C}}(B) \arrow[d, "\eta_B"] \\
GD \arrow[r, "G((\eta_B \circ h)^\sharp)"'] & G(D')
\end{tikzcd}
$$

*Taking* $F(A) = D, F(B) = D'$ *we obtain a functor* $F : \mathbf{C} \to \mathbf{D}$ *called* **free functor with respect to G**, *and a natural transformation* $\eta : I_{\mathbf{C}} \dot{\to} G \circ F$, *called* **universal transformation**, *such that* $F(h) = (\eta_B \circ h)^\sharp$.

**Proposition A.6.4** *Free functors are unique up to natural isomorphism.*

**Proposition A.6.5** *Free functors preserve colimits.*

**Proposition A.6.6** *For every set* $P$, $P^\oplus$ *is the free commutative monoid generated by* $P$. *This construction extends to a free functor* $(-)^\oplus : \mathbf{Set} \to \mathbf{CMon}$ *with respect to the forgetful functor* $V : \mathbf{CMon} \to \mathbf{Set}$, *where* $\mathbf{CMon}$ *is the category of free commutative monoids.*

## References


[1] J. Adamek, H. Herrlich, and G. Strecker. *Abstract and Concrete Categories.* Series in Pure and Applied Mathematics. John Wiley and Sons, 1990.

[2] M. Barr and C. Wells. *Category Theory for Computing Science.* Series in Computer Science. Prentice Hall International, London, 1990.

[3] R. M. Burstall and J. A. Goguen. Putting theories together to make specifications. In *Proc. Int. Conf. Artificial Intelligence*, 1977.

[4] H. Ehrig and B. Mahr. *Fundamentals of Algebraic Specification 1: Equations and Initial Semantics*, volume 6 of *EATCS Monographs on Theoretical Computer Science.* Springer, Berlin, 1985.

[5] C. Ermel. Kategorielle Darstellung von Stellen/Transition-Netzen. Studienarbeit – TU Berlin, December 1994.

[6] S. Mac Lane. *Categories for the Working Mathematician*, volume 5 of *Graduate Texts in Mathematics.* Springer, New York, 1971.

[7] A. Martini, H. Ehrig, and D. Nunes. Elements of basic category theory. Technical Report 96-5, Technical University Berlin, March 1996.

[8] J. Meseguer and U. Montanari. Petri nets are monoids. *Information and Computation*, 88(2):105–155, 1990.

[9] J. Padberg. *Abstract Petri Nets: A Uniform Approach and Rule-Based Refinement*. PhD thesis, Technical University Berlin, 1996.

[10] J. Padberg, H. Ehrig, and L. Ribeiro. Algebraic high-level net transformation systems. *Mathematical Structures in Computer Science*, 5:217–256, 1995.

[11] J. L. Peterson. Petri nets. *Computing Surveys*, 9(3):223–252, September 1977.

[12] J. L. Peterson. *Petri Net Theory and the Modeling of Systems*. Prentice-Hall, Englewood Cliffs, N. J., 1981.

[13] C.A. Petri. *Kommunikation mit Automaten*. PhD thesis, Schriften des Institutes für Instrumentelle Mathematik, Bonn, 1962.

[14] W. Reisig. *Petri nets*. Springer Verlag, 1985.

[15] L. Ribeiro, H. Ehrig, and J. Padberg. Formal development of concurrent systems using algebraic high-level nets and transformations. In *Proc. VII Simpósio Brasileiro de Engenharia de Software*, pages 1–16, Rio de Janeiro, oct. 1993. Tech-report no. 93-13, TU Berlin.

[16] L. Ribeiro and J. Padberg. Algebraic high-level nets and transformations with initial markings. Technical Report 94/7, 1994.