The Dimensions of Petri Nets:
The Petri Net Cube*

Ekkart Kindler       Michael Weber

Humboldt-Universität zu Berlin
Institut für Informatik
Unter den Linden 6
D-10099 Berlin
Germany†

Abstract

There exist many different Petri net formalisms. In this paper, we present the Petri Net Cube which helps to structure and classify the variety of Petri net formalisms. We show, that three basic aspects are sufficient for describing most classical Petri net formalisms. Since these aspects are independent of each other, we call them the Dimensions of Petri Nets.

1 Introduction

Petri nets have been introduced about 30 years ago. From that time on, many extensions and variants have been proposed. Today, there exist innumerable different formalisms of Petri nets.

In spite of these many different formalisms there is a good understanding of what Petri nets are. There are only a few basic aspects which make up a Petri net—and lots of extensions. We will argue that three basic aspects are sufficient for covering the most prominent Petri net versions such as EN-Systems (Elementary Net Systems), P/T-Systems (Place/Transition-Systems), Coloured Petri Nets (CP-Nets), Algebraic Nets, P/E-Systems (Predicate/Event-Systems), and FIFO Nets. Moreover, we will show that these three aspects are independent of each other; i.e. each aspect can be dealt with separately. Therefore, we call these basic aspects the Dimensions of Petri Nets and we call the space set up by these dimensions the Petri Net Cube. Note that many differences between Petri net formalisms are purely syntactical. For example, there are different syntactic representations of markings or arc inscriptions. In this paper, we do not deal with these syntactical differences; rather, we concentrate on semantical differences. We

*Supported by the Deutsche Forschungsgemeinschaft (DFG) within the research project 'Petri Net Technology'.
†E-mail: {kindler,mweber}@informatik.hu-berlin.de
will informally motivate the three Dimensions of Petri Nets and we will mathematically formalize the domain of each dimension. Furthermore, we will show how each point of the Petri Net Cube corresponds to a particular Petri net formalism.

The main contributions of the Petri Net Cube are the following:

1. The Petri Net Cube helps to understand Petri nets—their differences and their common ground. Since the dimensions have an immediate intuitive meaning they will serve didactic purposes.

2. The Petri Net Cube helps to structure and classify the variety of Petri net formalisms. Each formalism can be associated with a point in the Petri Net Cube.

3. The Petri Net Cube helps to unify analysis techniques. Some analysis techniques depend on one dimension, only; then, an analysis technique can be formalized for a complete plane of the Petri Net Cube and need not be formalized for each Petri net formalism separately.

In this paper, we concentrate on the first two aspects. We informally motivate the three dimensions in Sect. 2. In Sect. 3, we briefly introduce the algebraic prerequisites that are necessary for a formal definition of the Petri Net Cube and its dimensions in Sect. 4.

In the literature, there have been some approaches to extend, unify, or generalize Petri nets in an algebraic or categorical setting (e.g. [Bra92, Pad96]). We are not interested in generalizing Petri nets (as e.g. in [Bra92]), but we are interested in a clear presentation of the Dimensions of Petri Nets by identifying three independent parameters. Parameterized Petri nets were first proposed in [Pad96] for unifying different types of Petri nets. Here, we are interested in a generic firing rule which covers all Petri net formalisms of the Petri Net Cube. The dimensions formalize in which way Petri net formalisms may vary; the existence of a generic firing rule proves that all Petri net formalisms still have something in common.

2 Motivation

First, let us recall some well-known Petri net formalisms. These Petri net formalisms will be used to identify the basic Dimensions of Petri Nets. The common feature of all Petri net formalisms is the net structure, which consists of places, transitions, and arcs, whereas the differences are in the markings and arc inscriptions.

**Elementary Net Systems:** In an EN-System [Thi87, Roz87] a marking of a place may take two values: valid or not valid. Since validity is usually represented by a black token on the place, the two values are often called marked and unmarked.

**P/T-Systems:** In contrast to EN-Systems, the marking of a place in a P/T-System [Rei87, Lau87] is a natural number, which can be represented by a corresponding
number of black tokens on that place. The arcs of a P/T-System may be inscribed by a natural number; this number tells how many black tokens are removed or added by the occurrence of the corresponding transition.

**Coloured Petri Nets:** Up to now, we have only considered Petri nets with black tokens. In CP-Nets [Jen92], different (distinguishable) tokens on places are allowed. Figuratively, there are tokens with different colours. Then, a marking is a collection of such tokens, where multiple presence of the same token is allowed\(^1\). The tokens added or removed by the occurrence of a transition are represented by arc-inscriptions. However, the same transition may occur in several modes. Therefore, the inscription is a functional object which associates a collection of tokens to each mode.

**Algebraic Nets:** Algebraic Nets [Krä85, Bil89, Rei91] are a special version of Coloured Petri Nets, where the allowed token domain is specified by algebraic techniques. From the semantical point of view, there seems to be no difference—at first glance. For technical reasons, (most versions of) Algebraic Nets have one special semantical feature, so-called fixed *arc-weights* which means that the number of tokens added or removed by one arc is the same for all modes. Note that this is not required in Coloured Petri Nets.

**P/E-Systems:** Predicate/Event-Systems [Rei85] also allow distinguishable tokens. However, multiple presence of the same token is not allowed. This way, places can be considered as dynamically changing predicates.

**FIFO-Nets:** FIFO-Nets [Rou87] also allow distinguishable tokens. However, the marking of a place is a sequence of tokens rather than an unordered collection. Tokens are removed at one end of the sequence and are added at the other end. This way, a place behaves like a FIFO-channel.

Now, what are the basic differences of these Petri net formalisms and what are their common grounds? First, the formalisms differ in the kind of tokens which are allowed: Black tokens which cannot be distinguished from each other or coloured tokens which can be distinguished. This difference will be formalized by the dimension *token structure*.

Second, the formalisms differ in the kind of collections of tokens which are allowed on a place; technically speaking, we have sets (EN-Systems and P/E-Systems), multisets (P/T-Systems, Coloured Petri Nets), and sequences (FIFO-Nets). This difference will be formalized by the dimension *marking structure*.

Last, there is the subtle difference between Algebraic Nets and Coloured Nets concerning the flow of tokens through arcs. Each arc of an Algebraic Net has *fixed throughput*; an arc of a Coloured Petri Net may have *flexible throughput*. This difference will be formalized by the dimension *flow structure*.

In Sect. 4, we will formalize the domain of each dimension and we will show how a combination of values of each domain makes up a Petri net formalism. Since values of

\(^1\)Technically speaking, the marking is a multiset of tokens.
different dimensions can be combined arbitrarily with each other, the name ‘dimension’ is justified.

3 Algebraic prerequisites

For a formal definition of the Petri Net Cube, we need some algebraic concepts. In particular, we use monoids and free constructions with respect to a class of monoids.

In order to be self-contained, we briefly rephrase these concepts. We will define three classes of algebras \( \mathfrak{M}, \mathfrak{C}, \text{ and } \mathfrak{J} \). For understanding the basic idea of the Petri Net Cube, it is sufficient to know that \( \text{Free}(\mathfrak{M}, X) \) denotes the sequences over \( X \), to know that \( \text{Free}(\mathfrak{C}, X) \) denotes the multisets over \( X \), and to know that \( \text{Free}(\mathfrak{J}, X) \) denotes the sets over \( X \).

Algebras

An algebra \( A = (A, O) \) consists of a base set \( A \) and a sequence of operations \( O = (f_1, \ldots, f_n) \) on \( A \). Each operation \( f_i \) is a mapping \( f_i : A^{k_i} \to A \) where \( k_i \) is called the arity of \( f_i \). The sequence of arities \( (k_1, \ldots, k_n) \) is called the type of algebra \( A \).

For convenience, we do not distinguish a 0-ary operation \( f : A^0 \to A \) from an element of \( A \) and we call it a constant. For example, we write \( f \in A \) instead of \( f() \in A \). Moreover, we use infix notation for binary operations wherever appropriate.

Monoids

An algebra \( A = (A, (e, \oplus)) \) of type \( (0, 2) \) is a monoid, if we have

\[
\begin{align*}
x \oplus (y \oplus z) &= (x \oplus y) \oplus z \\
e \oplus x &= x \oplus e = x
\end{align*}
\]

for each \( x, y, z \in A \). This means that \( \oplus \) is associative and \( e \) is a neutral element.

Homomorphisms

For two algebras \( A = (A, (f_1, \ldots, f_n)) \) and \( B = (B, (g_1, \ldots, g_n)) \) of the same type, a mapping \( h : A \to B \) is a homomorphism from \( A \) to \( B \) if for each operation \( f_i : A^{k_i} \to A \) of \( A \) and each \( x_1, \ldots, x_{k_i} \in A \) we have \( h(f_i(x_1, \ldots, x_{k_i})) = g_i(h(x_1), \ldots, h(x_{k_i})) \). If \( h \) is a homomorphism from \( A \) to \( B \) we also write \( h : A \to B \).

Free algebras

In this paper, we consider classes of algebras (denoted by \( \mathfrak{A} \)) which satisfy particular properties; e.g. the class of all monoids or the class of all commutative monoids.

Now, suppose we have some set \( X \) and we want to have an algebra of \( \mathfrak{A} \) which at least contains the elements of \( X \) and which is “typical” (see [EM85] for details) for class \( \mathfrak{A} \). This can be achieved by the free algebra.
Let $\mathfrak{A}$ be a class of algebras and let $X$ be some set. An algebra $A = (A, O_A) \in \mathfrak{A}$ along with some mapping $u : X \to A$ is called free over $X$ in $\mathfrak{A}$ if for each algebra $B = (B, O_B) \in \mathfrak{A}$ and each mapping $h : X \to B$ there exists a unique homomorphism $\overline{h} : A \to B$ such that for each $x \in X$ we have $\overline{h}(u(x)) = h(x)$.

The free algebra $A$ is uniquely determined (up to isomorphism) by $\mathfrak{A}$ and $X$, if it exists at all. We say that the free construction exists for a class of algebras, if for any set $X$ there exists a free algebra. We denote this algebra by $\text{Free}(\mathfrak{A}, X)$.

**Examples**

In the rest of this paper we consider three classes of algebras: the class of monoids $\mathfrak{M}$, the class of commutative monoids $\mathfrak{C}$, and the class of idempotent and commutative monoids $\mathfrak{I}$. A monoid $A = (A, (e, \oplus))$ is commutative if for each $x, y \in A$ we have

$$x \oplus y = y \oplus x$$

A monoid $A$ is idempotent, if for each $x \in A$ we have

$$x \oplus x = x$$

For some set $X$, the free algebra $\text{Free}(\mathfrak{M}, X)$ gives us the sequences over $X$ with the empty sequence $\langle \rangle$ as its neutral element and concatenation $\circ$ as its binary operation. The free algebra $\text{Free}(\mathfrak{C}, X)$ gives us the multisets over $X$ with the empty multiset $\langle \rangle$ as neutral element and multiset sum $+$ as its binary operation. The free algebra $\text{Free}(\mathfrak{I}, X)$ gives us the sets over $X$ with the empty set $\{\}$ as its neutral element and union $\cup$ as its binary operation.

**Liftings**

We will use free constructions over some class of algebras $\mathfrak{A}$ for formalizing markings of a single place of a Petri net (cf. [MM90]). Then, the marking of a net with places $P$ is a vector $M : P \to \text{Free}(\mathfrak{A}, X)$. The neutral element $e$ of $\text{Free}(\mathfrak{A}, X)$ and the binary operation $\oplus$ immediately lift to vectors:

- $e : P \to \text{Free}(\mathfrak{A}, X)$ with $e(p) = e$ for each $p \in P$
- for $M_1, M_2 : P \to \text{Free}(\mathfrak{A}, X)$ we define $M_1 \oplus M_2$ by $(M_1 \oplus M_2)(p) = M_1(p) \oplus M_2(p)$ for each $p \in P$.

**4 The Petri Net Cube**

Now, we are ready for a formal presentation of the Petri Net Cube. We will first formally define its dimensions, i.e. the range of allowed values for each dimension. Then, we will define the concept of a Petri net type and we will formalize the instances of each Petri net type.
4.1 The Petri net square

For didactic reasons, we start with the formalization of only two dimensions: token structure and marking structure. The third dimension will be introduced in Sect. 4.2.

Token structure

As mentioned above, the dimension token structure determines the legal token domains of a Petri net type. This can easily be formalized by a class of sets. Each class of sets TS is a legal value of dimension token structure. Then, in a concrete Petri net of this type each \( A \in TS \) is a legal token domain.

Let us consider the values of TS for some well-known Petri net formalisms:

- \( TS_1 = \{\mathbb{1}\} \) is the value of dimension token structure which corresponds to EN-systems and P/T-systems.
- \( TS_2 = \{ A \mid A \text{ is a set} \} \) is the value which corresponds to coloured Petri nets and algebraic Petri nets.

Marking structure

The dimension marking structure describes which kind of collections of tokens make up a marking of a particular Petri net type. In Sect. 3, we saw that sequences, multisets, and sets can be formalized as a free algebra in some class of monoids. Other kinds of collections can be formalized by using other classes of monoids.

So, each class of monoids \( \mathfrak{M} \) for which the free construction exists is a legal value of the dimension marking structure. For example:

- \( MS_2 = \mathfrak{C} \), i.e. the class of commutative monoids, is the value corresponding to P/T-systems, Coloured Petri Nets and Algebraic Petri Nets, since \( \text{Free}(\mathfrak{C}, X) \) yields a multiset over \( X \).
- \( MS_1 = \mathfrak{M} \), i.e. the class of all monoids, is the value corresponding to FIFO-nets.
- \( MS_3 = \mathfrak{I} \), i.e. the class of commutative and idempotent monoids, is the value corresponding to EN-systems and P/E-systems.

Petri net types and instances

Given a legal value TS of the dimension token structure and a legal value MS of the dimension marking structure, we call the pair \( (TS, MS) \) a (two dimensional) Petri net type.

An instance of a Petri net type \( (TS, MS) \) consists of a net\(^2\) \( N = (P, T, F) \), a token domain \( A \in TS \), and a mapping \( i : F \to (A \to \text{Free}(MS, A)) \). This mapping associates a function to each arc; for a given mode \( a \in A \) this function evaluates to a collection of

\(^2\)N = (P, T, F) is a net, if P and T are disjoint sets and \( F \subseteq (P \times T) \cup (T \times P) \); the elements of P are called places, the elements of T are called transitions, and the elements of F are called arcs.
tokens which will be added or removed by the occurrence of the corresponding transition in mode \(a\). We call \((N, A, i)\) a Petri net (instance) of type \((TS, MS)\) and write \((N, A, i) : (TS, MS)\). A marking of \((N, A, i) : (TS, MS)\) is a vector \(M : P \rightarrow \text{Free}(MS, A)\). For some marking \(M \) of \((N, A, i)\) we call \(\Sigma = (N, A, i, M)\) a system of type \((TS, MS)\) and also write \(\Sigma : (TS, MS)\).

**Firing rule**

Next, we will define the firing rule for all Petri net types in a generic way. Of course, this definition must coincide with the firing rules for all classical Petri net versions.

The idea of the definition is quite simple. Let us consider a Petri net instance \((N, A, i) : (TS, MS)\), a transition \(t\), an occurrence mode \(a \in A\), and two markings \(M_1\) and \(M_2\) of \((N, A, i)\). First, we define two markings \(-t_a\) and \(t_a^+\) of \((N, A, i)\) which represent the tokens consumed or produced by the occurrence of transition \(t\) in mode \(a\):

\[
-t_a(p) = \begin{cases} i(p, t)(a) & \text{for } (p, t) \in F \\ e & \text{for } (p, t) \not\in F \end{cases}
\]

and

\[
t_a^+(p) = \begin{cases} i(t, p)(a) & \text{for } (t, p) \in F \\ e & \text{for } (t, p) \not\in F \end{cases}
\]

where \(e\) denotes the neutral element in Free\((MS, A)\).

Now, \(t\) is enabled in mode \(a\) at marking \(M_1\) if there exists a marking \(M\) of \((N, A, i)\) such that \(M_1 = M \oplus -t_a\), where \(\oplus\) denotes the lifting of the binary operation \(\oplus\) of Free\((MS, A)\) to markings. Informally, marking \(M_1\) is split into two parts: \(-t_a\) represents the tokens removed by the occurrence of transition \(t\) in mode \(a\); \(M\) represents all other tokens which are not affected by the occurrence of transition \(t\) in mode \(a\). Now, we can remove the tokens of \(-t_a\) and add the tokens of \(t_a^+\) which gives us \(M_2 = t_a^+ \oplus M\). Note that we deliberately remove tokens from the right-hand side and add them to the left-hand side; this gives us FIFO-behaviour for non-commutative \(\oplus\) operations.

Now, it seems to be straightforward to define the firing rule as follows: \(M_1 \xrightarrow{t} M_2\) iff there exists a marking \(M\) such that \(M_1 = M \oplus -t_a\) and \(M_2 = t_a^+ \oplus M\). This definition, however, has a flaw: For marking structure \(\mathcal{J}\) (i.e. for sets) it does not give us the classical firing rule for EN-systems. The reason is that by \(\{\bullet\} = \{\bullet\} \cup \{\bullet\}\) we can use a single token \(\bullet\) twice: once for removing it by the firing and once for leaving it unaffected by the firing of the transition. Figure 1 shows a behaviour that is legal under the above

![Figure 1: Wrong firing](image)

definition for a net of type \((TS_1, \mathcal{J})\). Therefore, the above definition does not capture the behaviour of EN-systems.
Similarly, firing would be possible in contact situations as indicated in Fig. 2. In order
to fix this problem, we require that the splittings $M_1 = M \oplus t_a$ and $M_2 = t_a^+ \oplus M$ are
disjoint splittings. To this end, we require that there does not exist a “smaller” marking
$M'$ with $M_1 = M' \oplus t_a$. Technically, we require that there are no two markings $M', M''$
such that $M = M' \oplus M''$, $M_1 = M' \oplus t_a$ and $M'' \neq \emptyset$. Altogether, we get the following
firing rule:

**Definition 1 (Generic firing rule)** Let $(N, A, i)$ be a Petri net instance of type
$(\text{TS}, \text{MS})$, let $t$ be a transition of $N$, let $a \in A$ be a mode, and let $M_1$ and $M_2$
be two markings of $(N, A, i)$. We define $M_1 \xrightarrow{t_a} M_2$ iff there exists a marking $M$
such that $M_1 = M \oplus t_a$ and $M_2 = t_a^+ \oplus M$ and there do not exist two markings $M'$
and $M'' \neq \emptyset$ such that $M = M' \oplus M''$ and $M_1 = M' \oplus t_a$ or such that $M = M'' \oplus M'$
and $M_2 = t_a^+ \oplus M'$.

### 4.2 The third dimension

Two dimensions are not sufficient for distinguishing between certain Petri net
formalisms. For example, we cannot distinguish Coloured Petri Nets and Algebraic Petri
Nets. Moreover, in a Petri net instance of type $(\text{TS}_1, \text{MS}_2)$ which almost resembles
P/T-system, arcs can be inscribed by $0$, which is rather unusual. In order to cope
with these mismatches, we introduce the third dimension flow structure, which restricts
arc-inscriptions.

**Cardinality**

The restrictions of the arc-inscriptions concern the number of tokens which may flow
through an arc. For example, in order to exclude the inscription $0$, we require $|i(f)(x)| \neq
0$ for each arc $f \in F$ and each $x \in A$, where $|$ denotes cardinality of sets. Since the
dimension flow structure should be independent of the other dimensions, we do not know
whether cardinality refers to sets, multisets or some other collection. Therefore, we need
a concept of cardinality which works for all free constructions.

Now, given some class of monoids $\mathfrak{A}$ and some domain $X$, how can we define the
cardinality of an element $m \in \text{Free}(\mathfrak{A}, X)$? In the following definition we will exploit
the fact that each element of $\text{Free}(\mathfrak{A}, X)$ can be generated from elements of $X$ and $e$.
Essentially, the cardinality of an element is the minimal number of operations needed
for this construction.

The formalization requires some technical effort: Let $\mathfrak{M}$ denote the class of all mon-
oids; then $\text{Free}(\mathfrak{M}, X)$ is well-defined and yields the sequences over $X$. We define a
mapping \( c : X \to \mathbb{N} \) by \( c(x) = 1 \) for each \( x \in X \). Since the algebra \( \mathcal{N} = (\mathbb{N}, 0, +) \) is a monoid and \( \text{Free}(\mathcal{M}, X) \) is the free construction, we know that there is a unique homomorphism \( \overline{c} : \text{Free}(\mathcal{M}, X) \to \mathcal{N} \) such that \( \overline{c}(x) = c(x) \) for each \( x \in X \) (indeed, \( \overline{c} \) gives us the length of a sequence). For any class of monoids \( \mathcal{A} \) for which the free construction exists, we know that \( \text{Free}(\mathcal{A}, X) \) is a monoid (i.e. is an element of \( \mathcal{M} \)). Let \( u \) be the universal mapping \( u : X \to \text{Free}(\mathcal{A}, X) \). Then, we know that there is a unique homomorphism \( \overline{u} : \text{Free}(\mathcal{M}, X) \to \text{Free}(\mathcal{A}, X) \) such that \( \overline{u}(x) = u(x) \) for each \( x \in X \).

Since \( \overline{c} \) and \( \overline{u} \) are uniquely defined, we can employ these homomorphisms for defining the cardinality for each element of \( m \in \text{Free}(\mathcal{A}, X) \) by \( \text{card}(m) = \min \{ \overline{c}(s) \mid s \in \text{Free}(\mathcal{M}, X) : \overline{u}(s) = m \} \)

**Defining the dimension** By help of the cardinality function, it is easy to formalize fixed or positive arc-weights: Fixed arc-weights can be expressed by \( \text{card}(i(f)(x)) = \text{card}(i(f)(y)) \) for each arc \( f \) and all modes \( x \) and \( y \). Positive arc-weights can be expressed by \( \text{card}(i(f)(x)) > 0 \). Since we always assume that the restriction applies to each arc, we will replace \( i(f) \) by \( i \) from now on. Then, a flow-condition is an expression constructed from terms \( \text{card}(i(x)), \text{card}(i(y)) \), natural numbers, and the relations = and >.

Any set of flow-conditions is a legal parameter of dimension flow structure. We define:

- \( FS_1 = \{ \} \) which does not restrict the flow structure
- \( FS_2 = \{ \text{card}(i(x)) > 0 \} \) for positive arc-weights
- \( FS_3 = \{ \text{card}(i(x)) = \text{card}(i(y)) \} \) for fixed arc-weights.

### 4.3 Petri net types and instances

Given a legal value \( TS \) of dimension token structure, a legal value \( MS \) of dimension marking structure, and a legal value \( FS \) of dimension flow structure, we call the triple \( (TS, MS, FS) \) a **Petri net type**. An instance \( (N, A, i) \) of the two dimensional Petri net type \( (TS, MS) \) is an instance of \( (TS, MS, FS) \), if for each arc \( f \in F \) and for each two modes \( x, y \in A \) each flow-condition of \( FS \) holds true.

Since each instance of \( (TS, MS, FS) \) is also an instance of the two dimensional Petri net type \( (TS, MS) \), we need not redefine the firing rule. The dimension flow structure does not restrict the firing rule, it just restricts arc inscriptions.

Note that there may be further dimensions which would require a redefinition of the firing rule. But these dimensions are beyond the scope of this paper.

### 5 Classification of Petri nets

The three dimensions establish a space which we call the **Petri Net Cube**. Each combination of the parameters given in Sect. 4 determines a more or less useful Petri net type.

Figure 3 shows the space of the Petri Net Cube. Each Petri net formalism described in
Sect. 2 is represented as a point or an edge in the Petri Net Cube as shown in Fig. 3. The following list gives the coordinates of some well-known Petri net formalisms in the Petri Net Cube. Note that we do not specify the third dimension for FIFO-nets because there are different versions of FIFO-nets in the literature; therefore, these different versions of FIFO-nets are represented as an edge of the Petri Net Cube.

- EN-systems: (TS₁, MS₃, FS₂)
- P/T-systems: (TS₁, MS₂, FS₂)
- Coloured Petri nets: (TS₂, MS₂, FS₁)
- Algebraic Petri nets: (TS₂, MS₂, FS₃)
- P/E-systems: (TS₂, MS₃, FS₂)
- FIFO-nets: (TS₂, MS₁)

6 Conclusion

In this paper, we have identified three basic aspects which allowed us to characterize most of the classical Petri net formalisms. We have formalized these aspects as independent dimensions of the Petri Net Cube. The definition of a generic firing rule shows that the concept of independent dimensions for classifying Petri nets does really work.

Of course, some Petri net formalisms are not yet covered by the Petri Net Cube. For example, inhibitor arcs or timing constraints are still missing. These aspects can be addressed by defining additional dimensions. Currently, we are investigating a dimension concerning timing aspects. The step taken in this paper from two to three dimensions
illustates that adding dimensions to the Petri Net Cube is possible. In this paper, we did not even need to change the generic firing rule when adding the third dimension. For more sophisticated extensions, however, a redefinition of the generic firing rule seems to be necessary.

Sometimes we would like to have two different versions of places within a single Petri net; for example, we would like to have FIFO-places and ordinary places. The idea presented in this paper can be extended to cope with such nets. Basically, we need to choose a value for dimension marking structure for each place individually. Choosing values for each dimension for each place individually would also allow different token domains for different places. The main purpose of this paper was to present the idea of the Petri Net Cube and to give it a semantical foundation. Therefore, we do not discuss these extensions here; these extensions are subject of further research and forthcoming papers.

Acknowledgements We would like to thank S. Gürses, S. Haar, T. Vesper, and H. Völzer for their helpful comments on this paper. Furthermore, we want to thank the ‘Kaffeerunde’ with members of W. Reisig’s group at Humboldt University for discussions on the Petri Net Cube.

References


