General Morphisms of Petri Nets*
(Extended Abstract)

Marek A. Bednarczyk and Andrzei M. Borzyszkowski

Institute of Computer Science, Gda
sk Branch, Polish Acad. of Sc.
Abrahama 18, 81-825 Sopot, Poland,  http://www.ipipan.gda.pl

Abstract. A new notion of a general morphism of Petri nets is introduced. The new morphisms are shown to properly include the morphisms considered so far. The resulting category of Petri nets is shown to admit products. Potential applications of general morphisms are indicated.

1 Introduction

For mathematically oriented people Petri nets are quite complex objects. The following observation should put the above statement into a proper perspective: it took a quarter of a century from the inception of Petri nets, cf. [12], to the definition of their morphisms, cf. [14][15].

Winskel’s solution to the problem of defining a suitable notion of Petri net morphism was algebraic. He noticed that Petri nets can be viewed as certain 2-sorted algebras. Consequently, Petri net morphisms defined in [14] are homomorphisms of the corresponding algebras.

The dynamic behaviour of a (marked) Petri net \( N \) is described by means of its case graph. Surely, if the case graph of a net is to be taken as the abstract representation of the dynamic behaviour of the net, then Petri net morphisms should give rise to morphisms of transition systems. More formally, the construction of the case graph of a net should be the object part of a functor from any category of Petri nets to the category of labelled transition systems described above. Indeed, the notion proposed by Winskel does satisfy the above criterion.

Some years later another class of morphisms was distinguished in an attempt to describe a coreflection between a category of elementary net systems and a category of elementary transition systems, cf. [11]. The morphisms introduced by Nielsen et al., form a subclass of Winskel morphisms (well, not quite, this is the price to pay for disallowing isolated places in nets). As such they also satisfy the criterion.

The existence of the case graph functor prompts a natural question. Namely, is it possible to find a converse construction? Such construction should produce a Petri net which implements a given transition system. The first results achieved in that direction are due to Ehrenfeucht and Rozenberg, cf. [8], where the

* Partially supported by LoSSeD workpackage within the CRIT-2 project funded by ESPRIT and INCO programmes, and by the State Committee for Scientific Research grants 8 T11C 018 11 and 8 T11C 037 16.

© Springer-Verlag Berlin Heidelberg 1999
so called *synthesis problem* is solved. The problem is to construct a Petri net, the case graph of which is isomorphic to a given transition system, and such that the transitions of the net are the labels of the transition system. The solution proposed in the above seminal paper hinges on the idea of a *region* in a transition system. With this notion one can characterize elementary transition systems as those which satisfy two regional separability conditions. Ehrenfeucht and Rozenberg showed that the case graphs of a class of Petri nets are elementary. Conversely, the regions of an elementary transition system taken as places provide a simple synthesis of a net.

This construction was generalized later in various ways. From our point of view it is important that some of these generalizations attempted to make the constructions functorial, cf. [11,6]. Thus, implicitly, the idea was to synthesize not only Petri nets from transition systems, but also Petri net morphisms from morphisms of the underlying transition systems.

So far, all functors were based on the idea that all regions should become places of the net constructed. The nets synthesized in this way are literally saturated with places. The wealth of places to choose from means that even a rather restricted subclass of Winskel’s morphisms is sufficient to synthesize all morphisms between elementary transition systems.

Sometimes, though, for instance for readability sake, it is preferable to construct a net with a small number of places. Unfortunately, Winskel morphisms are too demanding to allow synthesis of morphisms even in simple cases.

The general problem: to synthesize a Petri net that realizes a given concurrent behaviour, has already received a good deal of attention, see e.g., [1] for an in depth discussion. But, to the best of our knowledge, functoriality was discussed only for constructions resorting to saturated nets.

The reason seems pretty simple. Within the framework of categories of Petri nets considered until now, constructions other than those returning saturated nets are not going to be functorial.

Here, as a remedy, we propose a new class of *general* morphisms and its subclass of *rigid* morphisms. It turns out that Winskel’s morphisms are general. The new categories of general Petri nets and their *labelled* counterparts are shown to admit products.

Also, there are enough general morphisms to make the construction of a labelled state machine out of a transition system functorial. This simple observation has important consequences. In a companion paper, see [3], the authors show how the notions and the results presented here can be used to develop a functorial synthesis procedure for a wide class of concurrent behaviours. This procedure works for a class of asynchronous systems, cf. [13,2], namely those which can be presented as rigid or mixed product of (sequential) transition systems, see [7,16,10]. In [4] we show that the rigid product of Petri net realizations of a family of transition systems is a realization of the rigid product of the family. Thus, in the light of the results presented here, it is indeed sufficient to provide just a functorial realization of sequential systems.