

Generalized Petri Nets

Jade Master

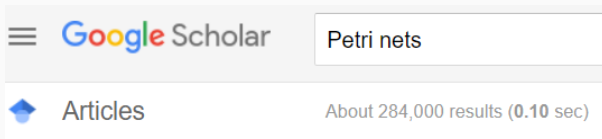
jmast003@ucr.edu

Nov 9, 2019

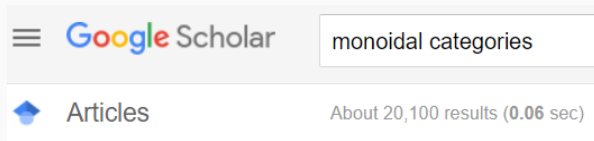
University of California Riverside

Q-Nets

There is a lot of work which has been done on Petri nets.



For comparison, if we search for the phrase "Monoidal Categories"



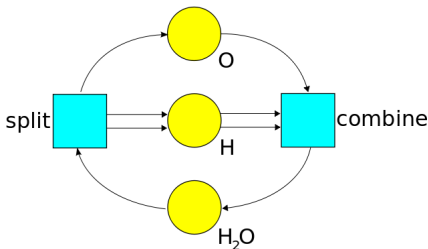
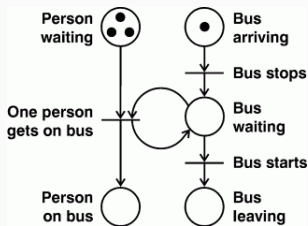
Many people have a specific application in mind.

Category theory is good at organizing mathematics.

Definition: A Petri net is a pair of functions of the following form

$$T \begin{matrix} \xrightarrow{s} \\ \xleftarrow{t} \end{matrix} \mathbb{N}[S]$$

where $\mathbb{N}: \text{Set} \rightarrow \text{Set}$ is the free commutative monoid monad which sends a set X to $\mathbb{N}[X]$ the free commutative monoid on X .



Petri Nets and String Diagrams

There are two ways that Petri nets relate to string diagrams:

- Petri nets *are* string diagrams. They are morphisms in symmetric monoidal categories with extra bells and whistles.

Petri Nets and String Diagrams

There are two ways that Petri nets relate to string diagrams:

- Petri nets *are* string diagrams. They are morphisms in symmetric monoidal categories with extra bells and whistles.
- Petri nets *generate* symmetric monoidal categories whose morphisms are string diagrams. More on this later.

Lawvere Theories

Definition: A Lawvere theory is category with finite products generated by a single object 1 . The objects can be thought of as natural numbers n with product given by $+$.

These should be thought of as platonic ideals of algebraic gadgets.

Example: The Lawvere theory \mathbf{MON} for monoids has morphisms

$$m: 2 \rightarrow 1$$

$$e: 0 \rightarrow 1$$

subject to associativity and unitality. A monoid is given by a product preserving functor

$$F: \mathbf{MON} \rightarrow \mathbf{Set}$$

We can replace \mathbb{N} in the definition of Petri net with a different monad. In 1963 Linton showed a correspondence between Lawvere theories and finitary monads on \mathbf{Set} .

$$\begin{array}{ccc} Q & & M_Q \\ \downarrow f & \mapsto & \Downarrow M^f \\ R & & M_R \end{array}$$

$M_Q X =$ the free model of Q on X

Definition: Let $\mathbf{Q}\text{-Net}$ be the category where

- objects are **Q-nets**, i.e. pairs of functions of the form

$$T \begin{smallmatrix} \xrightarrow{s} \\ \xrightarrow{t} \end{smallmatrix} M_Q S$$

- a morphism from the Q-net $T \begin{smallmatrix} \xrightarrow{s} \\ \xrightarrow{t} \end{smallmatrix} M_Q S$ to the Q-net

$$T' \begin{smallmatrix} \xrightarrow{s'} \\ \xrightarrow{t'} \end{smallmatrix} M_Q S' \text{ is a pair of functions}$$

$(f: T \rightarrow T', g: S \rightarrow S')$ such that the following diagrams commute:

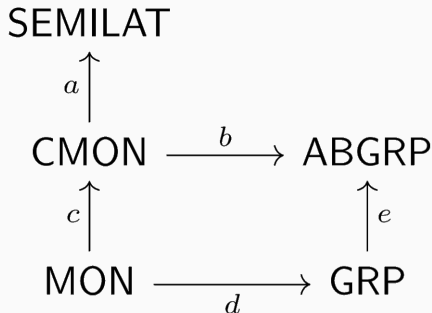
$$\begin{array}{ccc} T & \xrightarrow{s} & M_Q S \\ f \downarrow & & \downarrow M_Q g \\ T' & \xrightarrow{s'} & M_Q S' \end{array}$$

$$\begin{array}{ccc} T & \xrightarrow{t} & M_Q S \\ f \downarrow & & \downarrow M_Q g \\ T' & \xrightarrow{t'} & M_Q S' \end{array}$$

Q-Net extends to a functor

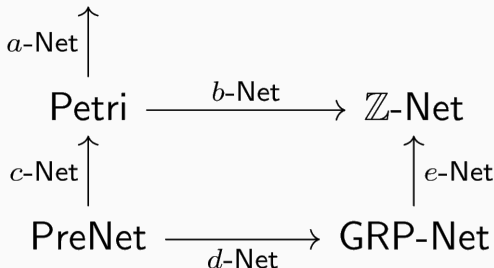
$$(-) - \text{Net} : \text{Law} \rightarrow \text{Cat}$$

where Cat is the category of small categories and functors. We can take the following diagram of Lawvere theories



to get the following network of categories which allows us to explore the relationships between different kinds of Q-nets.

SEMILAT-Net



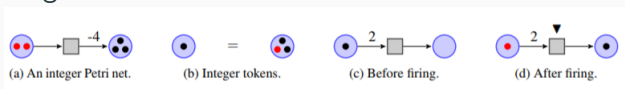
Many of these are familiar.

Many of these are familiar.

- PreNet is the category of pre-nets: Petri nets equipped with an ordering on the input and output of each transition. These are useful for generating processes in a way which keeps track of the identities of various tokens.

Many of these are familiar.

- PreNet is the category of pre-nets: Petri nets equipped with an ordering on the input and output of each transition. These are useful for generating processes in a way which keeps track of the identities of various tokens.
- \mathbb{Z} -Net is the category of integer nets studied in [3] and [4]. These are useful for modeling the concept of credit and borrowing.

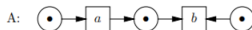


Many of these are familiar.

- PreNet is the category of pre-nets: Petri nets equipped with an ordering on the input and output of each transition. These are useful for generating processes in a way which keeps track of the identities of various tokens.
- \mathbb{Z} -Net is the category of integer nets studied in [3] and [4]. These are useful for modeling the concept of credit and borrowing.



- SemiLat-Net is the category of elementary net systems. These are Petri nets which can have a maximum of one token in each place. These are useful for modeling non-concurrent processes.



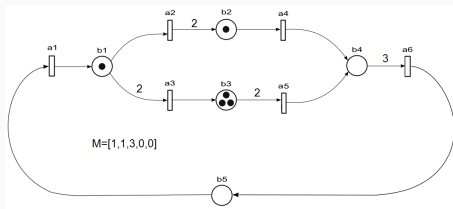
Generalized Semantics

Generalized Semantics

Petri nets are useful because they are a general language for representing processes which can be performed in sequence and in parallel. This can be summarized with following slogan:

Petri nets present free symmetric monoidal categories

Objects are given by possible markings and morphisms represent all possible ways to shuffle the markings around using the transitions.



Commutative monoidal categories

The devil is in the details.

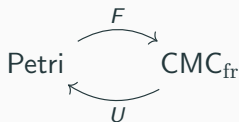
Because Petri nets have a free commutative monoid of species, they more naturally present **commutative monoidal categories**.

These are commutative monoid objects in \mathbf{Cat} .

$$\mathrm{Mor}C \begin{array}{c} \xrightarrow{s} \\ \xleftarrow{t} \end{array} \mathrm{Ob}C$$

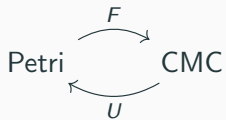
MacLane's coherence theorem doesn't apply.

- In *Petri Nets are Monoids* Messeguer and Montanari introduced the idea [1]. They construct a functor



where CMC is the category of commutative monoidal categories and CMC_{fr} is the full subcategory of CMC whose objects are commutative monoidal categories with a free monoid of objects. The freeness of the objects of CMC_{fr} is chosen to match the free commutative monoid of places in a Petri net.

With some help, we managed to obtain the following.



If the definition of Q-net is any good, there should be a similar adjunction.

Theorem (JM)

For every Lawvere theory Q there is an adjunction

$$\begin{array}{ccc} & F_Q & \\ \text{Q-Net} & \xrightarrow{\quad} & \text{Mod}(Q, \text{Cat}) \\ & U_Q & \end{array}$$

where $\text{Mod}(Q, \text{Cat})$ is the category of models of Q in the category of categories.

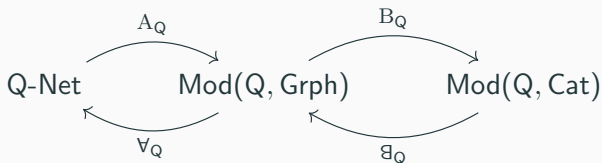
For a Q-net

$$P = T \underset{t}{\overset{s}{\rightrightarrows}} M_Q S$$

$F_Q(P)$ is the category where objects are given by $M_Q S$ and where morphisms are given by the free closure of T under the operations of Q and composition.

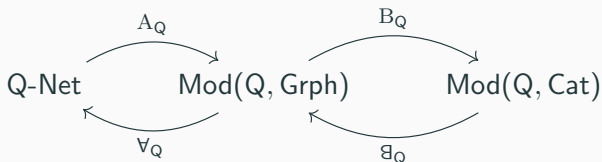
Proof: (sketch)

This adjunction can be factored into two parts.



Proof: (sketch)

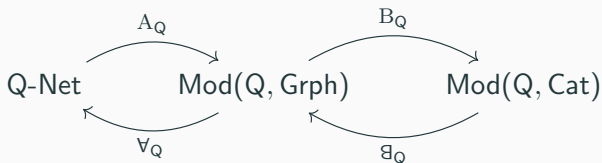
This adjunction can be factored into two parts.



- The first adjunction has a left adjoint which freely closes the transitions of your Q-net under the operations of \mathbf{Q} (Shulman).

Proof: (sketch)

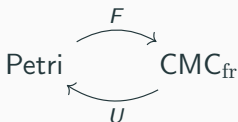
This adjunction can be factored into two parts.



- The first adjunction has a left adjoint which freely closes the transitions of your Q-net under the operations of Q (Shulman).
- The second adjunction has a left adjoint which freely closes the transitions under the operation of composition (Lack).

Putting Q-Nets to Use

The adjunction



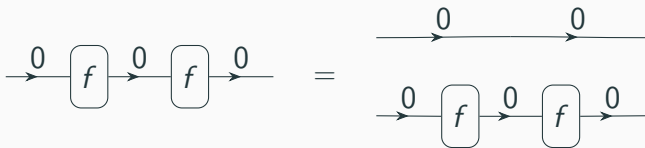
constructed in [1] wasn't entirely satisfactory to the Petri net community and our modification of it has the same pitfall. The slickest way to show that something is wrong is with the following proposition:

Theorem

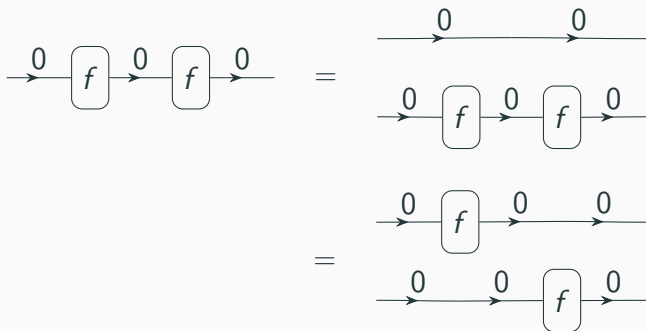
Let $f: 0 \rightarrow 0$ be a morphism in a commutative monoidal category C . Then

$$f \circ f = f + f$$

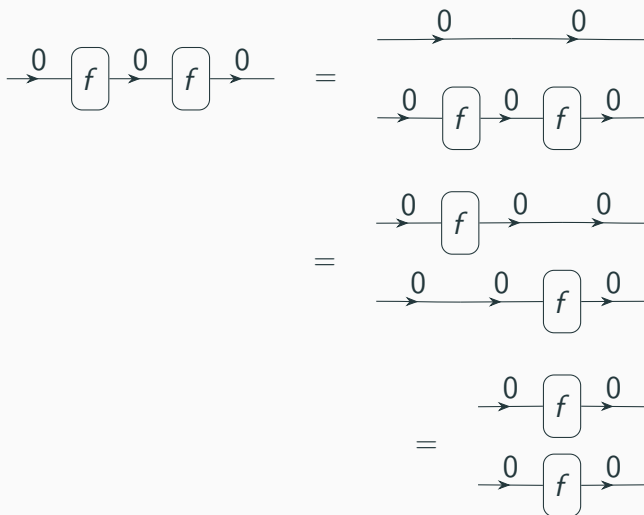
Proof.



Proof.



Proof.



□

So under appropriate conditions, parallel composition and sequential composition are the same even though they have very different semantic interpretations!

The fix is to make the categories not *strictly* commutative.

There were a few attempts to generate non-commutative symmetric monoidal categories from Petri nets. In 1994 Sassone constructed a pseudofunctor between the category of Petri nets and a category of non-strictly commutative symmetric monoidal categories. [2]

To get a free category which has some weak structure you should start with a Q-net which doesn't already have that property. To generate symmetric monoidal categories you should start with pre-nets. So far our structures give us the following two functors:

$$\begin{array}{ccc} & \text{Petri} & \\ & \uparrow & \\ c\text{-Net} & & \\ & \text{PreNet} & \xrightarrow{F_{\text{MON}}} \text{SMC} \end{array}$$

To get a free category which has some weak structure you should start with a Q-net which doesn't already have that property. To generate symmetric monoidal categories you should start with pre-nets. So far our structures give us the following two functors:

$$\begin{array}{ccc} & \text{Petri} & \\ & \uparrow & \\ c\text{-Net} & & \\ & \text{PreNet} & \xrightarrow{F_{\text{MON}}} \text{SMC} \end{array}$$

Composing with the "free strict symmetric monoidal category on a strict monoidal category" functor changes this to

$$\begin{array}{ccccc} & \text{Petri} & & & \\ & \uparrow & & & \\ c\text{-Net} & & & & \\ & \text{PreNet} & \xrightarrow{F_{\text{MON}}} \text{SMC} & \xrightarrow{N} & \text{SSMC} \end{array}$$

This suggests a method to get more expressive semantics out of Petri nets. There is a similar situation for integer nets.

$$\begin{array}{c}
 \mathbb{Z}\text{-Net} \\
 \uparrow \text{e-Net} \\
 \text{GRP-Net} \xrightarrow{F_{\text{GRP}}} \text{Mod}(\text{GRP}, \text{Cat}) \xrightarrow{K} \text{SCCC}
 \end{array}$$

where SCCC is the category of strict symmetric monoidal categories equipped with the structure of a group.

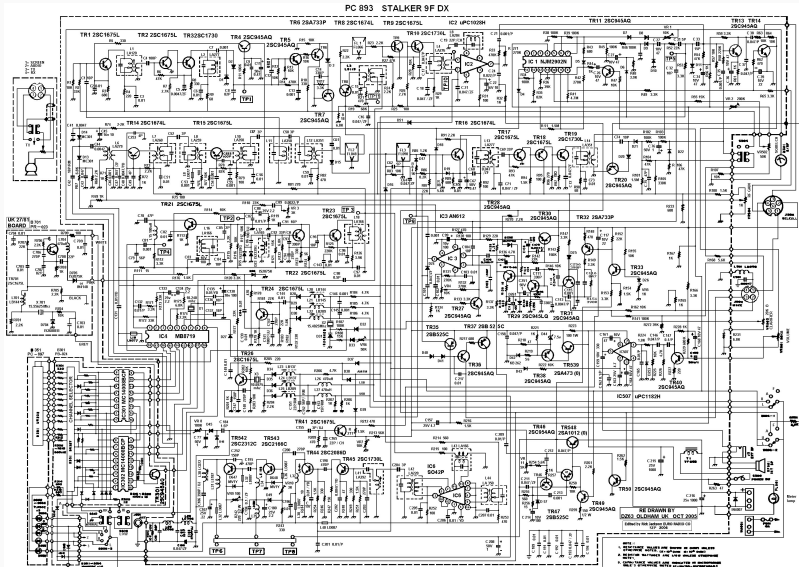
Petri nets are inherently categorical. There are many more opportunities for category theory to organize and understand the thousands of papers written on them.

- New types of nets. (e.g. let \mathcal{Q} to be the Lawvere theory for \mathbb{R}_+ modules or k -bounded nets).







Petri nets are inherently categorical. There are many more opportunities for category theory to organize and understand the thousands of papers written on them.

- New types of nets. (e.g. let \mathbf{Q} to be the Lawvere theory for \mathbb{R}_+ modules or k -bounded nets).
- Open \mathbf{Q} -nets. \mathbf{Q} -nets can be equipped with inputs and outputs so systems can be designed in a compositional way. This will extend the work of [6].

The Rising Sea



References

-  José Meseguer and Ugo Montanari (1990) Petri nets are Monoids
-  Vladimiro Sassone (1994) Strong Concatenable Processes: An Approach to the Category of Petri Net Computations
-  Fabrizio Genovese and Jelle Herold (2018) Executions in (Semi-)Integer Petri Nets are Compact Closed Categories
-  Massimo Bartoletti, Tiziana Cimoli, and G. Michele Pinna (2013) Lending Petri Nets
-  JM (2019) Generalized Petri Nets. Available at <https://arxiv.org/abs/1904.09091>
-  John Baez, JM (2018) Open Petri Nets. Available at <https://arxiv.org/abs/1808.05415>