Generalized Petri Nets

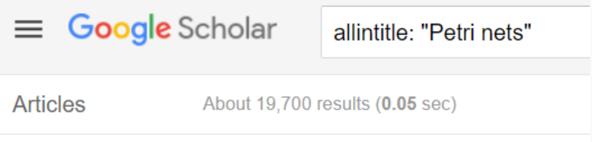
Jade Master
jmast003@ucr.edu

May 22, 2019

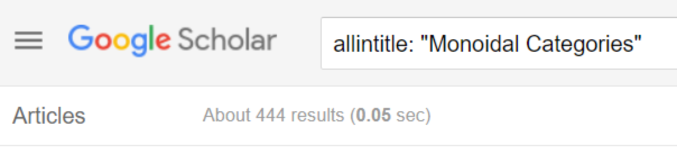
University of California Riverside

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"



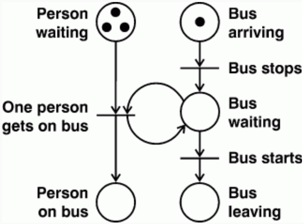
Many people have a specific application in mind.

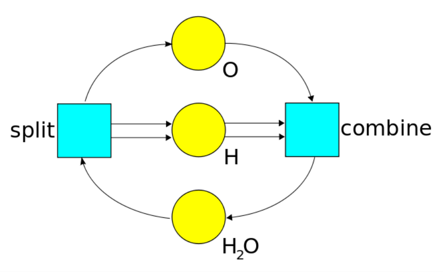
Category theory is good at organizing mathematics.

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

$$T \overset{s}{\underset{t}{\rightrightarrows}} \mathbb{N}[S]$$

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





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 MON for monoids has morphisms

$$m\colon 2 \to 1$$

$$e\colon 0 \to 1$$

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

$$F\colon \mathrm{MON} \to \mathrm{Set}$$

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 Set.

$$
\begin{tikzcd}
Q \arrow[d, "f"] & \mapsto & M_{\mathsf{Q}} \arrow[d, Rightarrow, "M^f"] \\
R & & M_{\mathsf{R}}
\end{tikzcd}
$$

$$M_{\mathsf{Q}}X = \text{ the free model of Q on } X$$

Definition: Let Q-Net be the category where

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

$$T \underset{t}{\overset{s}{\rightrightarrows}} M_{\mathrm{Q}} S$$

- a morphism from the Q-net $T \underset{t}{\overset{s}{\rightrightarrows}} M_{\mathrm{Q}} S$ to the Q-net $T' \underset{t'}{\overset{s'}{\rightrightarrows}} M_{\mathrm{Q}} S'$ is a pair of functions $(f \colon T \to T', g \colon S \to S')$ such that the following diagrams commute:

$$\begin{tikzcd} T \arrow[r, "s"] \arrow[d, "f"'] & M_{\mathrm{Q}} S \arrow[d, "M_{\mathrm{Q}} g"] \\ T' \arrow[r, "s'"'] & M_{\mathrm{Q}} S' \end{tikzcd} \qquad \begin{tikzcd} T \arrow[r, "t"] \arrow[d, "f"'] & M_{\mathrm{Q}} S \arrow[d, "M_{\mathrm{Q}} g"] \\ T' \arrow[r, "t'"'] & M_{\mathrm{Q}} S'. \end{tikzcd}$$

Q-Net extends to a functor

$$(-) - \mathsf{Net} \colon \mathsf{Law} \to \mathsf{Cat}$$

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

$$\begin{tikzcd}
\mathsf{SEMILAT} & \\
\mathsf{CMON} \arrow[u, "a"] \arrow[r, "b"] & \mathsf{ABGRP} \\
\mathsf{MON} \arrow[u, "c"] \arrow[r, "d"'] & \mathsf{GRP} \arrow[u, "e"']
\end{tikzcd}$$

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

$$
\begin{tikzcd}
\mathsf{SEMILAT\text{-}Net} & \\
\mathsf{Petri} \arrow[u, "a\text{-}\mathsf{Net}"] \arrow[r, "b\text{-}\mathsf{Net}"] & \mathbb{Z}\text{-}\mathsf{Net} \\
\mathsf{PreNet} \arrow[u, "c\text{-}\mathsf{Net}"] \arrow[r, "d\text{-}\mathsf{Net}"'] & \mathsf{GRP\text{-}Net} \arrow[u, "e\text{-}\mathsf{Net}"']
\end{tikzcd}
$$

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 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 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.

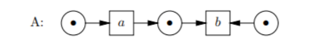
(a) An integer Petri net. (b) Integer tokens. (c) Before firing. (d) After firing.

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 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.

(a) An integer Petri net. (b) Integer tokens. (c) Before firing. (d) After firing.

- 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.



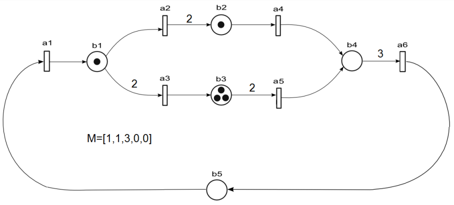
Generalized Semantics

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 Cat.

$$\mathrm{Mor}\, C \underset{t}{\overset{s}{\rightrightarrows}} \mathrm{Ob}\, C$$

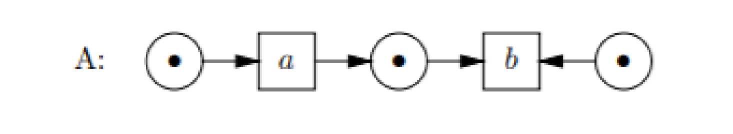
Maclane's coherence theorem doesn't apply.

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

$$\begin{tikzcd} \mathsf{Petri} \arrow[r, bend left, "F"] & \mathsf{CMC}_{\mathrm{fr}} \arrow[l, bend left, "U"] \end{tikzcd}$$

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

This wasn't entirely satisfactory to the Petri net community. The *individual token philosophy* vs. the *collective token philosophy*



The fix is to make the categories non-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]

With some help, we managed to obtain the following.

$$\begin{tikzcd} \mathrm{Petri} \arrow[r, bend left, "F"] & \mathrm{CMC} \arrow[l, bend left, "U"] \end{tikzcd}$$

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*

$$\begin{tikzcd} \mathsf{Q}\text{-}\mathsf{Net} \arrow[r, bend left, "F_{\mathsf{Q}}"] & \mathsf{Mod}(\mathsf{Q}, \mathsf{Cat}) \arrow[l, bend left, "U_{\mathsf{Q}}"] \end{tikzcd}$$

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

For a Q-net

$$P = \; T \underset{t}{\overset{s}{\rightrightarrows}} M_{\mathsf{Q}}S$$

$F_{\mathsf{Q}}(P)$ is the category where objects are given by $M_{\mathsf{Q}}S$ 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 three parts.

$$
\begin{tikzcd}
\mathsf{Q}\text{-}\mathsf{Net} \arrow[r, bend left, "\mathrm{A}_{\mathsf{Q}}"] & \mathsf{Q}\text{-}\mathsf{Net}_* \arrow[l, bend left, "\forall_{\mathsf{Q}}"] \arrow[r, bend left, "\mathrm{B}_{\mathsf{Q}}"] & \mathsf{Mod}(\mathsf{Q}, \mathsf{Grph}_*) \arrow[l, bend left, "ꓭ_{\mathsf{Q}}"] \arrow[r, bend left, "\mathrm{C}_{\mathsf{Q}}"] & \mathsf{Mod}(\mathsf{Q}, \mathsf{Cat}) \arrow[l, bend left, "Ɔ_{\mathsf{Q}}"]
\end{tikzcd}
$$

Proof: (sketch)

This adjunction can be factored into three parts.

$$\begin{tikzcd}
\mathsf{Q\text{-}Net} \arrow[r, bend left, "\mathrm{A}_{\mathsf{Q}}"] & \mathsf{Q\text{-}Net}_* \arrow[l, bend left, "\forall_{\mathsf{Q}}"] \arrow[r, bend left, "\mathrm{B}_{\mathsf{Q}}"] & \mathsf{Mod}(\mathsf{Q}, \mathsf{Grph}_*) \arrow[l, bend left, "\rotatebox[origin=c]{180}{B}_{\mathsf{Q}}"] \arrow[r, bend left, "\mathrm{C}_{\mathsf{Q}}"] & \mathsf{Mod}(\mathsf{Q}, \mathsf{Cat}) \arrow[l, bend left, "\rotatebox[origin=c]{180}{C}_{\mathsf{Q}}"]
\end{tikzcd}$$

- $$\mathrm{A}_{\mathsf{Q}} \dashv \forall_{\mathsf{Q}} \colon \mathsf{Q\text{-}Net} \to \mathsf{Q\text{-}Net}_*$$

  is the adjunction whose left adjoint freely adds an identity transition to every place.

Proof: (sketch)

This adjunction can be factored into three parts.

$$\begin{tikzcd}
\mathsf{Q\text{-}Net} \arrow[r, bend left, "\mathrm{A}_{\mathsf{Q}}"] & \mathsf{Q\text{-}Net}_* \arrow[l, bend left, "\forall_{\mathsf{Q}}"] \arrow[r, bend left, "\mathrm{B}_{\mathsf{Q}}"] & \mathsf{Mod}(\mathsf{Q}, \mathsf{Grph}_*) \arrow[l, bend left, "\rotatebox[origin=c]{180}{B}_{\mathsf{Q}}"] \arrow[r, bend left, "\mathrm{C}_{\mathsf{Q}}"] & \mathsf{Mod}(\mathsf{Q}, \mathsf{Cat}) \arrow[l, bend left, "\rotatebox[origin=c]{180}{C}_{\mathsf{Q}}"]
\end{tikzcd}$$

- $$\mathrm{A}_{\mathsf{Q}} \dashv \forall_{\mathsf{Q}} \colon \mathsf{Q\text{-}Net} \to \mathsf{Q\text{-}Net}_*$$

  is the adjunction whose left adjoint freely adds an identity transition to every place.

- $$\mathrm{B}_{\mathsf{Q}} \dashv \rotatebox[origin=c]{180}{B}_{\mathsf{Q}} \colon \mathsf{Q\text{-}Net}_* \to \mathsf{Mod}(\mathsf{Q}, \mathsf{Grph}_*)$$

  is the adjunction whose left adjoint freely closes the transitions under the operations of Q.

- The previous two adjunctions were constructed by hand. However, C_{Q} and Ↄ_{Q} are constructed with abstraction. There is a 2-functor

$$\mathsf{Mod}(\mathsf{Q}, -) \colon \mathsf{CAT}_{fp} \to \mathsf{CAT}$$

where CAT is the 2-category of categories and CAT_{fp} is the 2-category of categories with finite products, finite product preserving functors, and natural transformations. C_{Q} and Ↄ_{Q} are given by hitting the adjunction

$$\begin{tikzcd} \mathsf{Grph}_* \arrow[r, bend left, "L"] & \mathsf{Cat} \arrow[l, bend left, "R"] \end{tikzcd}$$

with $\mathsf{Mod}(\mathsf{Q}, -)$. $\blacksquare$

We can put our network of Q-nets to use. All of these have the collective token philosophy. To get a free category which has some weak structure you should start with a Q-net which doesn't already have that property.

$$
\begin{tikzcd}
\mathsf{Petri} & & \\
\mathsf{PreNet} \arrow[u, "c\text{-Net}"] \arrow[r, "F_{\mathrm{MON}}"] & \mathsf{SMC} \arrow[r, "N"] & \mathsf{SSMC}
\end{tikzcd}
$$

There is an analogous situation for integer nets.

$$\begin{tikzcd} \mathbb{Z}\text{-Net} \\ \text{GRP-Net} \arrow[u, "e\text{-Net}"] \arrow[r, "F_{\text{GRP}}"] & \text{Mod}(\text{GRP}, \text{Cat}) \arrow[r, "K"] & \text{SCCC} \end{tikzcd}$$

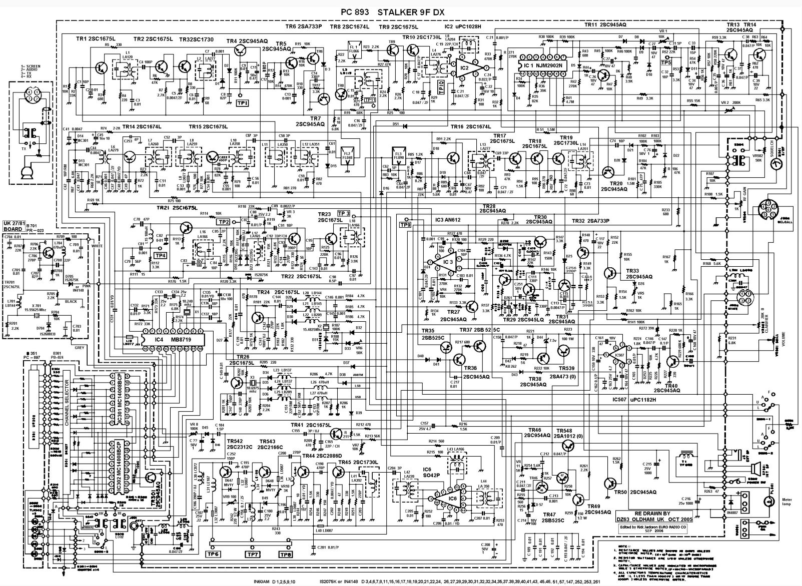
where SCCC is the category of strict symmetric monoidal categories equipped with the structure of a group.

Conclusion

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 Q to be the Lawvere theory for $\mathbb{R}_+$ modules).
- Open Q-nets. Q-nets can be equipped with inputs and outputs so systems can be designed in a compositional way. This extends the work of [6].

The end


PC 893 STALKER 9F DX
RE DRAWN BY
DZ63 OLDHAM UK OCT 2005
Edited by Rick Jackson EURO RADIO CO
SEP 2006

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`