CARTESIAN CLOSED CATEGORIES OF ROSEN'S (M, R)-SYSTEMS

FABRIZIO ROMANO GENOVESE AND FOSCO LOREGIAN

Abstract. We say everything, proving nothing.

1. Introduction

In the '50s the famous biologist Robert Rosen tried to propose a categorical model [**?**, **?**, **?**, **?**] for living systems based on the fundamental schematization of an 'organism' as a structure capable of *metabolic* functions and *repairing* functions. This is reasonable: at the very least, a living being is a connected region of space capable of gathering nutrients from its exterior, in order to transform them into energy devoted to work against entropic degeneration, re-generating parts of itself.

The second process can be understood in many ways, as 'rebuilding the enzymes that catalyse a reaction' or 'producing the necessary catalyst/reagents for a given reaction to start'; for example, the oxidation of glucose ($C_6H_{12}O_6$) in anaerobic conditions produces ATP and lactic acid as a result, as depicted in the following diagram.

$$\begin{tikzcd} & 2CH_3CH(OH)COOH \\ C_6H_{12}O_6 \arrow[rr] \arrow[ur, bend right] & & 2ATP \end{tikzcd} \tag{1.1}$$

On the other hand, in presence of water and some other catalysts, or in an acidic aqueous solution, ATP reacts with water to yield ADP and a phosphate:

$$\begin{tikzcd} & H_2O \arrow[d] & \\ ATP \arrow[r, no head] \arrow[dr, bend right] & {} \arrow[r] & ADP \\ & P_i & \end{tikzcd} \tag{1.2}$$

Evidently, the reactions can be 'composed'; this has motivated scientists like Petri to introduce a graphical representation for such processes.

For the sake of the present introduction, we call a 'reaction' pretty much anything that happens at a cellular, microscopic or mesoscopic level.

In modern terms, one can try to formalise the above intuition as follows:

Definition 1.1. Fix a cartesian closed category $\mathcal{K}$. Define a category $\mathsf{Enz} = \mathsf{Enz}(\mathcal{K})$ whose objects are types in $\mathcal{K}$ (regarding the latter as a model for λ-calculus), and morphisms are arrows $\zeta : R \longrightarrow P$ that we have to imagine as going from a tuple of reagents $(R_1, \dots, R_n)$ to a set of products (of the reaction catalyzed by ζ) $(P_1, \dots, P_n)$; of course this perspective lends itself to a fruitful formalisation: we are considering the cartesian multicategory where enzymes are *colors* and reactions are *multimorphisms*.

forse no, forse sta roba è lineare: in fin dei conti reagenti e prodotti si consumano.

Remark 1.2. A first problem that renders this approach quite artificial is: it is unclear how to distinguish a reaction that 'just happens, and everything stops' from a reaction that helps another reaction to keep going on.

Reactions have a *cost*, and thus they can exhaust the material that is necessary to ignite them, or they can just go wrong for some other reason (getting old, degenerative diseases...).

For this precise reason, Rosen proposed to refine this description assuming that together with reactions $\zeta : R \longrightarrow P$ we have 'helpers' or 'repairers' that allow the reaction $\zeta : R \longrightarrow P$ to continue.

This can evidently be rephrased elegantly in terms of a Petri net or a string diagram, where ζ is a certain graph that can be composed to others *à la* Petri nets, or alternatively, exploiting multi/polycategorically representing ζ as $\vec{R} \longrightarrow \vec{P}$.

In the present work we mostly concentrate on a different approach, exploiting the structure of Enz; we define a *repairer map* to be a morphism of the form

$$\Phi : P \longrightarrow [R, P] \tag{1.3}$$

for obects $R, P \in \mathsf{Enz}$, where $[R, P]$ denotes the internal hom in Enz, such that the following condition is satisfied:

> the map Φ sends a product $p \in P$ into a *unique* reaction $\zeta : R \longrightarrow P$ yielding the products p starting from some reagents $r \in R$.

(We will examine later the reason why Rosen makes this very strong assumption and a possible way to weaken it.)

Definition 1.3 (The category of $(\boldsymbol{M}, \boldsymbol{R})$-systems in $\mathcal{C}$)**.** Let $\mathcal{C}$ be a cartesian closed category; an (M, R)-system (of length 1) (f, Φ) consists of a diagram

$$A \xrightarrow{f} B \xrightarrow{\Phi} [A, B] \tag{1.4}$$

in $\mathcal{C}$, with the property that $\Phi(fa) = \ulcorner f \urcorner$, if $\ulcorner f \urcorner$ corresponds to f in the isomorphism $\mathcal{C}(1, [A, B]) \cong \mathcal{C}(A, B)$.

A morphism (u, v, w) of (M, R)-systems $(f : A \longrightarrow B, \Phi)$ and $(g : C \longrightarrow D, \Psi)$ consists of a triple of morphisms $u : A \longrightarrow C$, $v : B \longrightarrow D$ and $w : [A, B] \longrightarrow [C, D]$

with the property that the two squares in

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[d, "u"'] & B \arrow[r, "\Phi"] \arrow[d, "v"] & {[A,B]} \arrow[d, "w"] \\ C \arrow[r, "g"'] & D \arrow[r, "\Psi"'] & {[C,D]} \end{tikzcd} \tag{1.5}$$

are commutative.

With this definition, it is easily checked that the class of (M, R)-systems forms a category that we dub $(M, R)\mathsf{Sys}$.

We can extend the previous definition to allow longer tuples of objects in an (M, R)-system: the notion of morphism acts on components. This defines a category $\mathsf{g}(M, R)\mathsf{Sys}$ of *generalised* (M, R)-systems.

Definition 1.4 (General $\boldsymbol{(M, R)}$-system)**.** A *general* (M, R)-system $(\boldsymbol{f}, \boldsymbol{\Phi})$ consists of a tuple $\boldsymbol{f} = (f_1, \dots, f_n)$, a tuple $\boldsymbol{\Phi} = (\Phi_1, \dots, \Phi_n)$ with $f_i : A_i \longrightarrow B_i$ and $\Phi_i : \prod_j B_j \longrightarrow [A_i, B_i]$, organised in such a way that the tuple $\mathfrak{mr} = (\prod_i [A_i, B_i], \prod_i A_i, \prod_i B_i)$ is an automaton. This makes sense since the tuple $\boldsymbol{\Phi} = (\Phi_1, \dots, \Phi_n)$ determines a unique $\hat{\Phi} : \prod_j B_j \longrightarrow \prod_i [A_i, B_i]$ by the universal property of the product, and the tuple $\boldsymbol{f} = (f_1, \dots, f_n)$ determines a unique morphism $\hat{f} : \prod_i A_i \longrightarrow \prod_i B_i$.

The original definition of this category dates back to Rosen's [**?, ?, ?, ?**], and it was independently re-discovered by Căzănescu in []; to Căzănescu we owe the proof that the category $\mathsf{g}(M, R)\mathsf{Sys}$ is well-behaved in various senses: in short, $\mathsf{g}(M, R)\mathsf{Sys}(\mathcal{C})$ is a complete and cocomplete cartesian closed category.

Moreover, (M, R)-systems (f, Φ) are particular automata with state space $[A, B]$, input and output sets given by the domain and codomain of f respectively; more precisely [Arbib, 1966], $(M, R)\mathsf{Sys}$ embeds in the category of automata where morphisms (u, v, w) of automata are restrained by the request that w, u are injective and v is bijective (?).

However, here we do not concentrate on $(M, R)\mathsf{Sys}$ or $\mathsf{g}(M, R)\mathsf{Sys}$ but instead on a new look on a category of infinite chains of repairers.

2. In terms of types

This particular second point of view lends itself to the following generalisation: Rosen considers a sequence of functions (a *metabolic system*)

$$A \xrightarrow{f} B \xrightarrow{\Phi_1} [A, B] \xrightarrow{\Phi_2} [B, [A, B]] \xrightarrow{\Phi_3} [[A, B], [B, [A, B]]] \longrightarrow \dots \tag{2.1}$$

inductively defined in a certain way.

The map f has to be interpreted as a certain kind of process transforming reagents A into products B; the map Φ_1 acts as a repairer for f, and more generally, declaring $\Phi_0 := f$, each map Φ_{n+1} acts as a repairer for the map Φ_n.

In an arrangement of objects and morphisms like this, a «self-sustained» system is one such that, considering the infinite sequence of processes-and-repairers $(f, \Phi_1, \Phi_2, \dots)$, the infinite system of equations

$$(\clubsuit) \qquad \begin{cases} fa = b \\ \Phi_1(fa) = f \\ \Phi_2(\Phi_1(fa))(fa) = f \\ \vdots \end{cases}$$

has as unique solution an initial element $a \in A$ making all the equations $(\clubsuit)$ true. chiaramenteva data una regola induttiva per le equazioni, cosa che a suo tempo mi sono dimenticato di fare.

The maps Φ_n can be regarded as 'helpers' for the reaction f, in the sense that given the initial $a \in A$, the infinite chain of equations above yields that $\Phi_1(b)$ is a function from a yields b, Φ_2 is a function that yields a function that yields a reaction that yieldd b from a, etc.

A few remarks are in order now: first, let's clarify the inductive definition of the system $(\clubsuit)$:

$$(2.2) \qquad \begin{cases} X_0 = A \\ X_1 = B \\ X_n = [X_{n-2}, X_{n-1}] \end{cases} \qquad \begin{cases} f : A \longrightarrow B \\ \Phi_n : X_n \nrightarrow X_{n+1}, \\ \Phi_n(x_n) := \{x_{n+1} \in X_{n+1} \mid \exists x_{n-1} \in X_{n-1} : x_{n+1}(x_{n-1}) = x_n\} \end{cases}$$

A purported solution of $(\clubsuit)$ shall 'inhabit a type such that $X \cong [X, X]$' (obtained passing to the limit in the defining recursive equation?); or it is more difficult to get such a thing?

This makes very clear a first point:

Remark 2.1. The "incredibly strong assumption" above can be replace by the waaay more intuitive and natural one: $\Phi_1 : B \nrightarrow [A, B]$ is a *relation* and (b, f) are in Φ_1 iff there exists an a such that $fa = b$ (so at least the initial reaction f must be a function!).

Hence, $\Phi_1(b)$ is (as it should be) the set of all f's that make a certain metabolic reaction possible. And a similar property defines all the others Φ_n.

The additiona question is, does a given metabolic system have a colimit?

Alas, at this level of generality, the answer is no: as Rel does not have sequential colimits [].

But not all is lost! The same naive construction that does not give the colimit, turns out to compute a *lax* colimit (lax adjoint coolimit in Milius paper; but I guess it's the good old lax bicolimit of [?])![1]

[1]It is reasonable that Rosen hadn't arrived as far as bicolimits! Only the univ prop became slightly more elusive. The general construction is in the paper by Milius.

Now the problem is at least framed correctly: one can look at solutions in Rel (and there will be a certain meaning for a 'lax transfinite composition'), or in Set (and there will be a more "classical", super-restrictive meaning for an element a such that equations (♣) have solution).

For what concerns the solution in Rel, Milius shows that one has to do the following iterated limit/colimit construction; for the sake of clarity, let's stick to the more general problem of having a sequence

$$X_0 \nrightarrow X_1 \nrightarrow X_2 \nrightarrow \dots \tag{2.3}$$

of morphisms in the bicategory Rel of sets and relations, and suppose we want to compute its 'colimit'. Then, representing said sequence as a sequence of spans *of functions*,

$$\begin{tikzcd}[column sep=small] & X_{01} \arrow[dl] \arrow[dr] & & X_{12} \arrow[dl] \arrow[dr] & & X_{23} \arrow[dl] \arrow[dr] & \\ X_0 & & X_1 & & X_2 & & \dots \end{tikzcd} \tag{2.4}$$

it is clear that we can perform the pullback of every $X_i \longleftarrow X_{i,i+1} \longrightarrow X_{i+1}$ and $X_{i+1} \longleftarrow X_{i+1,i+2} \longrightarrow X_{i+2}$ obtaining an object $X_{[i,i+2]}$; we can then iterate the construction 'one level higher', so to obtain

$$\begin{tikzcd}[column sep=small] & & \vdots \arrow[dl] & & \vdots \arrow[dl] & & & & \\ & X_{012} \arrow[dl] \arrow[dr] & & X_{123} \arrow[dl] \arrow[dr] & & X_{234} \arrow[dl] \arrow[dr] & & & \\ X_{01} \arrow[d] \arrow[dr] & & X_{12} \arrow[dl] \arrow[dr] & & X_{23} \arrow[dl] \arrow[dr] & & X_{34} \arrow[dl] & & \\ X_0 & X_1 & & X_2 & & X_3 & & & \dots \end{tikzcd} \tag{2.5}$$

Formally: we have an indexed set $\{X_S \mid S \in P_0(\mathbb{N})\}$, indexed by the finite subsets of $\mathbb{N}$ (secretly ordered); we consider the object

$$\begin{aligned} \bar{X}_i &:= \lim_{S \in P_0(\mathbb{N}_{>i})} X_{\{i\} \cup S} \\ &= \lim \big(X_i \longleftarrow X_{\{i,i+1\}} \longleftarrow X_{\{i,i+1,i+2\}} \longleftarrow \dots \big) \end{aligned}$$

and after this, the colimit of the chain

$$\bar{X}_0 \longrightarrow \bar{X}_1 \longrightarrow \bar{X}_2 \longrightarrow \dots \tag{2.6}$$

We legitimately call this object $\bar{X}_\infty$.

We have thus built the object 'iteratively'

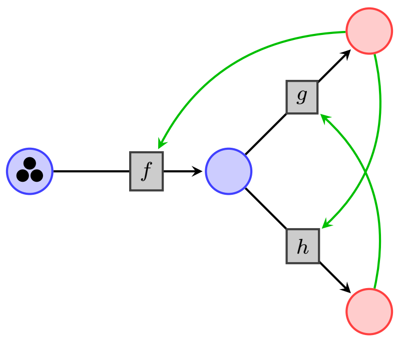
(2.7)
$$\begin{tikzcd}
 & & & & & \bar{X}_0 \arrow[ddddrrrrrr, bend left] & & \bar{X}_1 \arrow[ddddrrrr, bend left] & & & & \\
 & & & \vdots \arrow[dl] & & \vdots \arrow[dl] & & & & & & \\
 & & X_{012} \arrow[dl] \arrow[dr] & & X_{123} \arrow[dl] \arrow[dr] & & X_{234} \arrow[dl] \arrow[dr] & & & & & \\
 & X_{01} \arrow[dl] \arrow[dr] & & X_{12} \arrow[dl] \arrow[dr] & & X_{23} \arrow[dl] \arrow[dr] & & X_{34} \arrow[dl] & & & & \\
X_0 & & X_1 & & X_2 & & X_3 & & & \dots & & \bar{X}_\infty
\end{tikzcd}$$

3. Self-sustained systems

Bla bla continuano a venire fuori i punti fissi per descrivere i sistemi closed.

A system is closed in Rosen's sense if every component in the system except for a set of input components is maintained and every mapping in the system is implemented by a component within the system

Per esempio questo sistema è chiuso



(3.1)

Tale sistema si descrive con un diagramma in una CCC della forma

$$\begin{tikzcd}
 & & {[A,B] \times [B,Y] =: X} \\
A \arrow[r, "f"] & B \arrow[ur] \arrow[dr] & \\
 & & {[B,X] =: Y}
\end{tikzcd} \tag{3.2}$$

sicché X, Y si interdefiniscono tra loro; sostituendo opportunamente uno all'altro si ottengono le equazioni ai tipi

$$\begin{cases} X & \cong [A, B] \times [B, [B, X]] =: FX \\ Y & \cong [B, [A, B] \times [B, Y]] =: GY \end{cases} \tag{3.3}$$

so autopoiesis is a facet of self-recursion and interdefinition.

Questa costruzia determina X, Y come punti fissi degli endofuntori $F, G : \mathcal{C} \longrightarrow \mathcal{C}$ definiti da

$$FU = [A, B] \times [B, [B, U]] \qquad GV = [B, [A, B] \times [B, V]] \tag{3.4}$$

Se $\mathcal{C}$ è una categoria sufficientemente smooth, i funtori F, G sono accessibili (con rango determinato dal rango di presentabilità di A, B), sicché sappiamo che esiste un punto fisso [**?**, **?**] per la catena

$$\varnothing \longrightarrow F\varnothing \longrightarrow FF\varnothing \longrightarrow \cdots \tag{3.5}$$

ottenuta in κ iterazioni; notando che entrambi F, G sono funtori della forma $S \times [T, -]$ per certi insiemi S, T poi si riesce anche a stimare con precisione κ e a dare un'idea di come è fatto il pto fisso.

Remark 3.1. Quando i funtori sono puntati, che si fa?

3.1. **A general recipe.** Motivati dall'esempio precedente diamo la seguente

Definition 3.2 (Metabolic network)**.** Una metabolic network (*metnet* for short) è un ipergrafo i cui iperlati che non sono lati sono tutti della forma $v \longrightarrow [v_0, v_1]$.

Chiamando *2-limitato* un ipergrafo tale che tutti i suoi iperlati che non sono lati sono della forma $v \longrightarrow [v_0, v_1]$, dove v, v_0, v_1 sono vertici, una metnet è un iperdigrafo 2-limitato.

Now, data una metnet $\mathfrak{M}$ possiamo considerare la categoria cartesiana chiusa $\mathcal{C}(\mathfrak{M})$ che ha i vertici di $\mathfrak{M}$ come oggetti, e associare ad $\mathfrak{M}$ un diagramma in $\mathcal{C}(\mathfrak{M})$, dove ogni edge $U_0 \xrightarrow{f} U_1$ definisce un morfismo $\ulcorner f \urcorner : U_0 \longrightarrow U_1$, e ogni iperedge $V_{0i} \longrightarrow \{V_{1i}, V_{2i}\}$ prescrive che nel nodo V_{0i} venga messo l'hom interno $[V_{1i}, V_{2i}]$, con la convenzione che se da uno stesso V_0 si dipartono due iperlati $\{V_1, V_2\} \longleftarrow V_0 \longrightarrow \{V_1', V_2'\}$, al nodo V_0 del diagramma mettiamo il prodotto $[V_1, V_2] \times [V_1', V_2']$; quando questa cosa è stata fatta per ogni iperedge, il diagramma è pronto.

A questo punto, dal fatto che certi vertici saranno definiti in termini l'uno dell'altro, otteniamo delle eq di punto fisso. Le soluzioni di questo (sitema di) eq di punto fisso sono gli *stati self-sustained* della metnet.

Example 3.3. The initial example. Lo possiamo rappresentare come la metnet che segue:

$$\begin{tikzcd} A \arrow[r, "f"] & B \arrow[r, "g"] \arrow[d, "h"'] & C \\ & D & \end{tikzcd} \tag{3.6}$$

Ai tre iperlati associamo il diagramma (**??**); da qui otteniamo le equazioni di pto fisso (**??**), le cui soluzioni sono . . . .

Example 3.4. The diagram su whiteboard

$$\begin{tikzcd} A \arrow[r, bend left, "f"] \arrow[r, bend right, "f"'] & B \arrow[r, "g"] \arrow[d] & C \\ & D & \end{tikzcd} \tag{3.7}$$

agli iperlati associamo il diagramma

$$\begin{tikzcd} & {[A,B] \times [A,B]} \\ A \arrow[r, shift left] \arrow[r, shift right] & B \arrow[u] \arrow[d] \\ & {[B,[A,B] \times [A,B]]} \end{tikzcd} \tag{3.8}$$

che è già 'risolto'. Quindi questo sistema è già 'stabile' nello stato che lo inizia.

Example 3.5. from Chemero, A., Turvey, M.T., 2006. Complexity and "closure to efficient cause". In: Ruiz-Moreno, K., Barandiaran, R. (Eds.), ALIFE X: Workshop on Artificial Autonomy. MIT Press, Cambridge, MA, pp. 13–18.

$$\begin{tikzcd} A \arrow[r, "f"] & B & \\ & C & D \arrow[l, "g"] \end{tikzcd} \tag{3.9}$$

ai due iperlati associamo il diagramma

$$\begin{tikzcd} & A \arrow[r] & {[D,C] = B} \\ {C = [A,B]} & D \arrow[l] & \end{tikzcd} \tag{3.10}$$

da cui deduciamo che $C = [A, [D, C]]$ e $B = [D, [A, B]]$ (i due funtori a meno di iso sono uguali: sono $[A \times D, -]$).

Example 3.6. from Chemero, A., Turvey, M.T., 2008. Autonomy and hypersets. BioSystems 91, 320– 330.

$$
\begin{tikzcd}
 & Q & & & \\
P \arrow[rr, "f"'] & & A & & \\
 & R \arrow[rr, "g"'] & & B & \\
 & & S \arrow[rr, "h"'] & & C \\
 & & & T \arrow[uu, bend right] &
\end{tikzcd}
\tag{3.11}
$$

Questo è piu complicato: si ottiene il diagramma (sconnesso)

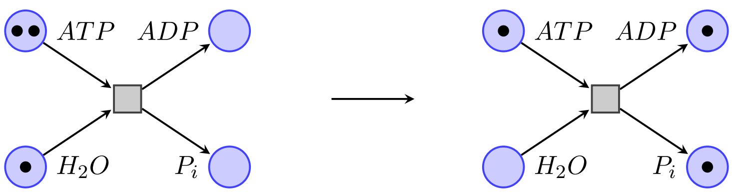
$$
\begin{tikzcd}
 & {[P,A]} & & & \\
P \arrow[rr, "f"'] & & \overset{A}{[R,B]} & & \\
 & R \arrow[rr, "g"'] & & \overset{B}{[S,C]} & \\
 & & S \arrow[rr, "h"] & & \overset{C}{[T,B]} \\
 & & & T &
\end{tikzcd}
\tag{3.12}
$$

da cui ricaviamo che

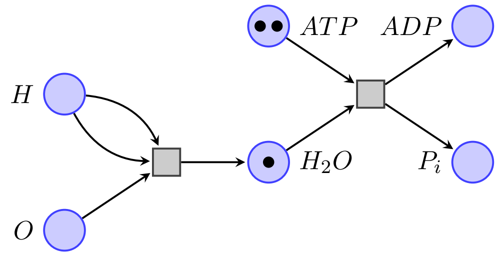
$$
\begin{cases} Q = [P, A] \\ A = [R, B] \\ C = [T, B] \end{cases} \qquad B = [S \times T, B] \tag{3.13}
$$

4. In terms of nets

Petri nets have been originally created with the purpose of representing chemical reactions. In this interpretation, places of the net represent chemical compounds such as atoms and molecules, with tokens signifying how much of any given resource the net has available. Transitions are reactions that turn compounds into other compounds.



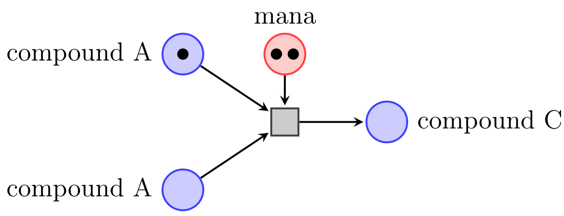
The formalism is very flexible. For instance, we could be more specific about how water is formed by adding another reaction to the net above:



Definition 4.1 (Petri net)**.** Denoting the set of *multisets* over a set S with $S^{\oplus}$, we define a *Petri net* as a couple functions $T \xrightarrow{s,t} S^{\oplus}$ for some sets T and S, called the set of places and transitions of the net, respectively.

Petri nets provide a framework which allows us easily phrase interesting questions in a formal way, such as "If we start with this distribution of compounds, will this reaction ever terminate? If so, which distribution of compounds will we have when the reaction ends?"

Still, this is not enough to represent biological processes à la Rosen. As Rosen posits, many interesting reactions in biochemistry are mediated by enzymes, that degrade over time. Following the terminology of the popular card game *Magic: The gathering*, we can represent the "viability" of reactions by endowing them with *mana*: Once a reaction is out of mana, it cannot fire anymore:



In Rosen terms, mana is an umbrella term that can represent enzymes, units of energy or catalysts of any sort that tend to degrade over time, for whatever reason. It is obvious that mana for a given reaction can be just represented as a standard place in a net, but this doesn't make it justice: We want to think about mana as a separated concept since we interpret it as a necessary ingredient to make a reaction work, to which we are not directly interested.

We can use the language of *categorical semantics for Petri nets* to axiomatise this idea precisely:

Definition 4.2 (Category of executions of a Petri net)**.** Let $N : T \xrightarrow{s,t} S^{\oplus}$ be a Petri net. A *commutative monoidal category* is a monoidal category with a commutative monoid of objects. From N, we can generate a *free commutative strict monoidal category (FCSMC)*, $\mathfrak{C}(N)$, as follows:

- The monoid of objects is $S^{\oplus}$. Monoidal product of objects A, B, denoted with $A \oplus B$, is given by the multiset sum;
- Morphisms are generated by T: each $u \in T$ corresponds to a morphism generator $su \xrightarrow{u} tu$; morphisms are obtained by considering all the formal (monoidal) compositions of generators and identities.

A detailed description of this construction can be found in [].

In this definition, objects represent markings of a net: $A \oplus A \oplus B$ means "two tokens in A and one token in B". Morphisms represent executions of a net, mapping markings to markings. A marking is reachable from another one iff there is a morphism between them.

Definition 4.3 (Non-local commutative semantics for a Petri net)**.** Let N be a Petri net and let $\mathcal{S}$ be a monoidal category. A *Petri net with a non-local commutative semantics* is a couple $\left(N, N^{\sharp}\right)$, with $N^{\sharp}$ a lax-monoidal functor $\mathfrak{C}(N) \longrightarrow \mathcal{S}$.

A morphism $\left(N, N^{\sharp}\right) \longrightarrow \left(M, M^{\sharp}\right)$ of Petri nets with commutative semantics is a strict monoidal functor $\mathfrak{C}(N) \xrightarrow{F} \mathfrak{C}(M)$ together with a natural transformation $N^{\sharp} \Rightarrow F \mathbin{⨟} M^{\sharp}$.

We denote the category of Petri nets with a non-local commutative semantics with $\mathbf{Petri}^{\mathcal{S}}$.

5. Systems with feedback loops: another view on fixed points

"... a completely autonomous living organism needs to encode all of the information about the state of all its catalysts, and, when necessary, make the necessary replacements itself." – This is not true.

I think a lot of this information is encoded in the interaction between a living system and it's environment. It's not that the system must encode all relevant information and make all relevant replacements, but that it must not make these replacements impossible. It's about constraints.

I think it is a very reasonable idea to think of living systems in terms of feedback loops, in particular fixed points, because both have the property of "emergent stability". I think this is a pretty popular idea, and dates at least from the 1920s, see diagram from von Uexkull: