ALife: The Nervous System as the 3-Pullback between the Brain and Physicality

Nick Rossiter
Computer Science
University of Northumbria
rossiternick1169@gmail.com
https://nickrossiter.org.uk/process/

Presented at

- ANPA 47
  - Alternative Natural Philosophy Association
- 5th August 2026
- University of Manchester

Abstract

In previous work on Artificial Life the nervous system as a whole was considered in categorical terms to be the connection between the neural networks of the brain and the physical actions of the body. In the current paper the nervous system will be explored in detail, looking at the Central Nervous System, which includes the brain and the spinal chord, and the Peripheral Nervous System, which controls the nerves in the rest of the body. Nerves are considered to involve both digital and analog components, the former considered to involve categories such as Cartesian with signals for pathways and the latter considered to involve continuous categories such as vector spaces with graded, or weighted signals. This work will expand the higher-order category theory developed at the end of last year's ANPA presentation.

Outline of Talk 1

- Recent work summarised
  - Current Status of AI
    - Much hyperbole
    - Increased functionality in weak AI
      - But it’s still weak AI
      - Distinction between success with artefacts and humans
  - Advances in category theory applications, useful in AI
    - Quantitative (weighted) arrows
    - Monoidal categories for internal structures, such as vectors
    - Higher-order n-categories with n-cells
    - Relaxation of associativity rules in higher-order categories

Outline of Talk 2

- Current status of categorial work on ALife
  - Use of 1-monads for handling physicality (physical processes)
  - Use of composed 2-monads for handling of layers in ANN (Artificial Neural Net for brain)
  - Example of ANN with image recognition to show grounding in real-world for part of model
  - Introduction of 3-pullback to represent relationship between brain and physical aspects of body with respect to the nervous system
  - Elaboration of 3-pullback as extension of work

Current Status of AI

Great Progress

- Weak AI
  - The ability to handle specific tasks like a human
  - Trained for specific tasks
- has advanced considerably in areas such as
  - Large language models
    - Queries, chats, drug research
  - Image recognition and manipulation
    - Medical scans, self-driving, verifying identity
  - Generative AI
    - New content from existing: document, image, coding
- All assisted by massive advances in processor capability
  - Nvidia’s parallel chips

GPAI/Regulation

- Is it approaching Strong (General Purpose) AI?
  - Human-like cognitive capabilities
  - Some claim it is, particularly from Anthropic, OpenAI
  - Major strengths in artefacts such as computer systems
  - But across human reasoning some doubts
- Whatever, Strong AI will require regulation:
  - Once we understand how to create a ‘being’ with Strong AI
    - We can elevate that comprehension to create Super AI (Sci-Fi AI)
  - There’s no reason to think that human intelligence (Strong AI) is a ceiling
  - The EU does have regulation; the USA does not.

Advances in Category Theory of Assistance to AI

Weights

- Important as provides quantitative category theory
- Lawvere metric space
  - can select real number to represent the distance between two points, with curvature permitted.
  - Topology context so non-Euclidean
  - Triangle inequality
  - Now viewed more as an example of an enriched category

Triangle Inequality

- The triangle inequality, in its most basic form, states that for any triangle, the sum of the lengths of any two sides must be greater than or equal to the length of the third side.
  - The distance d(x,z) between any two points x,z is no larger than the sum of the distances d(x,y) and d(y,z) via any third point y

$$
\begin{tikzcd}
 & y \arrow[dr, "g"] & \\
x \arrow[ur, "f"] \arrow[rr, "f+g"'] & & z
\end{tikzcd}
$$

Enriched Weighted Categories

- A weighted category is a category, where to each morphism $f: X \rightarrow Y$
    - Is assigned a non-negative real number (up to ∞)
        - the weight $w(f)$
- Take a short function $f: (X, w_X) \rightarrow (Y, w_Y)$
- A closed monoidal (enriched) category is constructed by taking the tensor product of weighted sets in the Cartesian product of the sets (X x Y):
    - The tensor product $X \otimes Y$ assigns to each pair <x,y> the weight $w_{(x \otimes y)}(x,y) = w_X(x) + w_Y(y)$

Monoidal Category 1 (Basics)

- A monoidal category is a category $\mathcal{C}$ equipped with
  - a functor such as $\otimes : \mathcal{C}\times\mathcal{C}\longrightarrow\mathcal{C}$
  - an object $1 \in \mathcal{C}$ called the unit object or tensor unit
  - a natural isomorphism $a_{x,y,z}:(x\otimes y)\otimes z \rightarrow x\otimes(y\otimes z)$ called the associator
  - a natural isomorphism $\lambda_x:1\otimes x \rightarrow x$ called the left unitor
  - a natural isomorphism $\rho_x:x\otimes 1 \rightarrow x$ called the right unitor

Monoidal Category 2 (Strict)

- Triangle identity
  - Any two tensor products are associative

$$\begin{tikzcd}
(x \otimes 1) \otimes y \arrow[rr, "a_{x,1,y}"] \arrow[dr, "\rho_x \otimes 1_y"'] & & x \otimes (1 \otimes y) \arrow[dl, "1_x \otimes \lambda_y"] \\
& x \otimes y &
\end{tikzcd}$$

- Pentagon identity
  - Any three tensor products are associative

Further facilities beyond Tensor Products

- Monoidal categories are fundamental building blocks for higher categories, which are used to model more complex mathematical structures.
  - Monoidal categories involve an extra layer of arrows for internal structure so arrow structure is more layered
- We need to move to category of category
  - Higher-order n-category theory

n-Cells

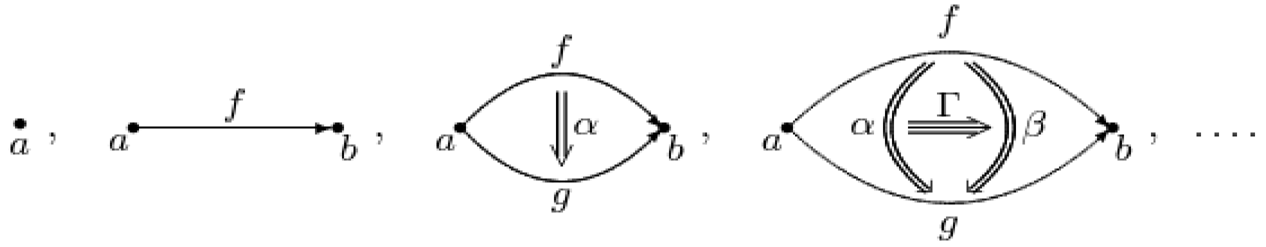
- Category of category or double category (n=2)
- We have met the concept before as adjunction
- It is also now known as a lens category
- It is a type of n-category, containing the basic cells:
  - 0-cell is objects (identity arrows, simple set)
  - 1-cell is arrows between objects (category, collection of functions)
  - 2-cell is arrows between arrows (natural transformation, adjunction)
  - 3-cell is (arrows between arrows) between (arrows between arrows)
    - Or adjunction arrows mapped onto adjunction arrows (modification)
    - We (Sisiaridis) used 3-cells for security in information systems
  - 4-cell and beyond have been used in knot theory

n-Category

- No upper limit to n, which is the highest ranking cell in the category
- A category with just 0-cells (objects) is a set, a discrete category
- A category with just 1-cells (arrows between objects) is an ordinary category
- A category holding up to 2-cells (arrows between arrows) is a 2-category
- A category holding up to 3-cells -- (arrows between arrows) between (arrows between arrows) -- is a 3-category
- And so on

Lens Diagram

Cell types



0-cell 1-cell 2-cell 3-cell

Isomorphism and Associativity

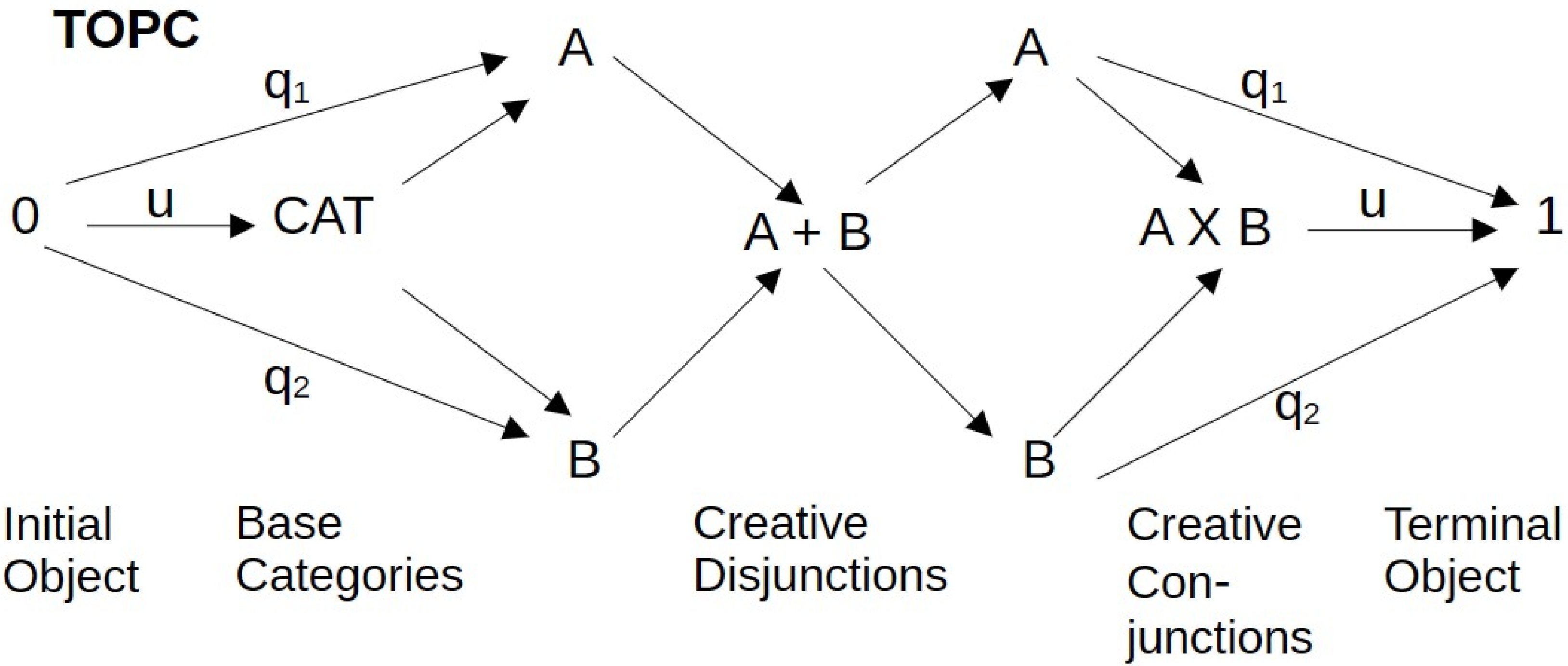
- In n-category (n>=2)
  - unique up to natural Isomorphism (composition) and associativity are no longer in general enforced
- Except in strong (strict) n-category
  - Composition and associativity are strictly applied throughout
  - These do not occur often in nature
- In weak n-category (multicategory)
  - Composition and associativity are restricted up to a particular level of isomorphism (2-isomorphisms for instance in a 3-category)

Categorial Representation of Artificial Life Physicality

Looked at Works of Rosen and Whitehead

- Rosen – Life Itself
  - Developed a formalism for natural systems using functors from category theory
- Whitehead – Process & Reality
  - Developed a philosophy for life which can be interpreted as informal category theory
- Using formal category theory
  - Developed their ideas into a topos as the basis of ALife
  - Close to Whitehead's Category of the Ultimate
  - A tension between union and product
  - Starting from an initial object 0, through base categories, through disjunctions of these base categories, through conjunctions of these disjunctions, to a terminal object 1
  - A Topos shown in canonical form as TOPC

ALife



TOPC
0
u
CAT
q1
q2
A
B
A + B
A
B
A X B
q1
q2
u
1
Initial Object
Base Categories
Creative Disjunctions
Creative Con-junctions
Terminal Object

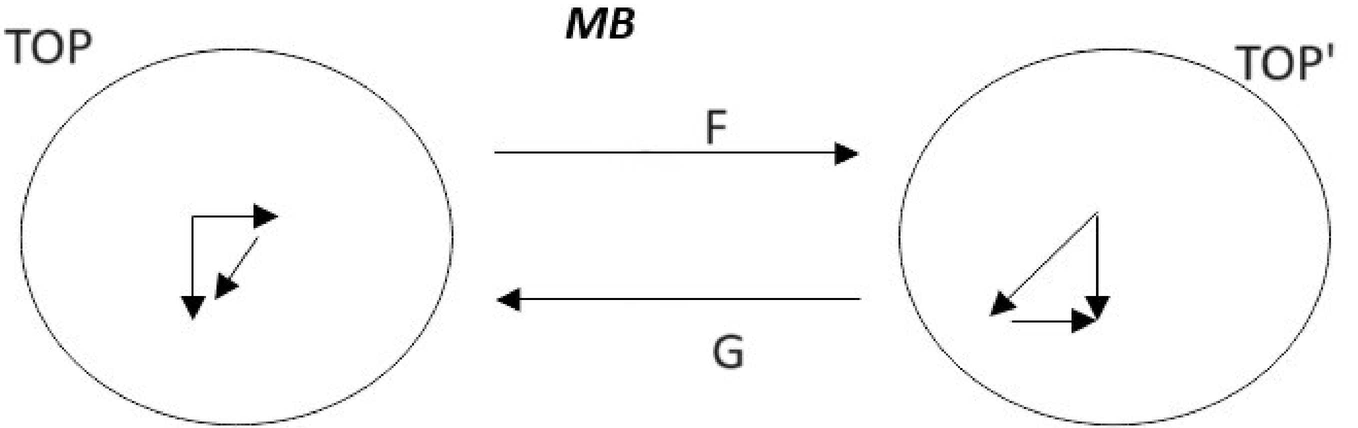
Process as a Monad

- We used a monad to represent transactional process on a Cartesian closed structure with certain ‘nice’ properties such as limits and a subobject classifier
  - A topos
- The monad represented activity transactionally through an adjointness between current state and next state
- The topos represented data structure

Process on the Topos

- Taking the topos TOP from one state to another TOP’
- A free functor F takes it forward in creativity
- An underlying functor G takes it back in checking viability
- There is adjointness $F \dashv G$ if certain conditions hold
- The whole is a Monad **MB** (1-Monad) on the adjointness
- The Monad is the Process of ALife

Monad **MB** for Topos Transition: The Adjunction $F \dashv G$



Use of composed 2-monads for handling of layers in ANN (Artificial Neural Net for brain)

Theories for ANNs

- Inspired by biology
  - Nodes with connections
  - Connections have weights
  - Weights are adjusted in training
- In training
  - Universal function approximator
    - Approximate any continuous function to an arbitrary degree of accuracy
    - Learn and represent a vast range of functions
- Deep learning – multiple hidden layers
- Output of each neuron computed by some non-linear function of the totality of its inputs
- Output is a real number

Artificial Neural Nets (ANN)

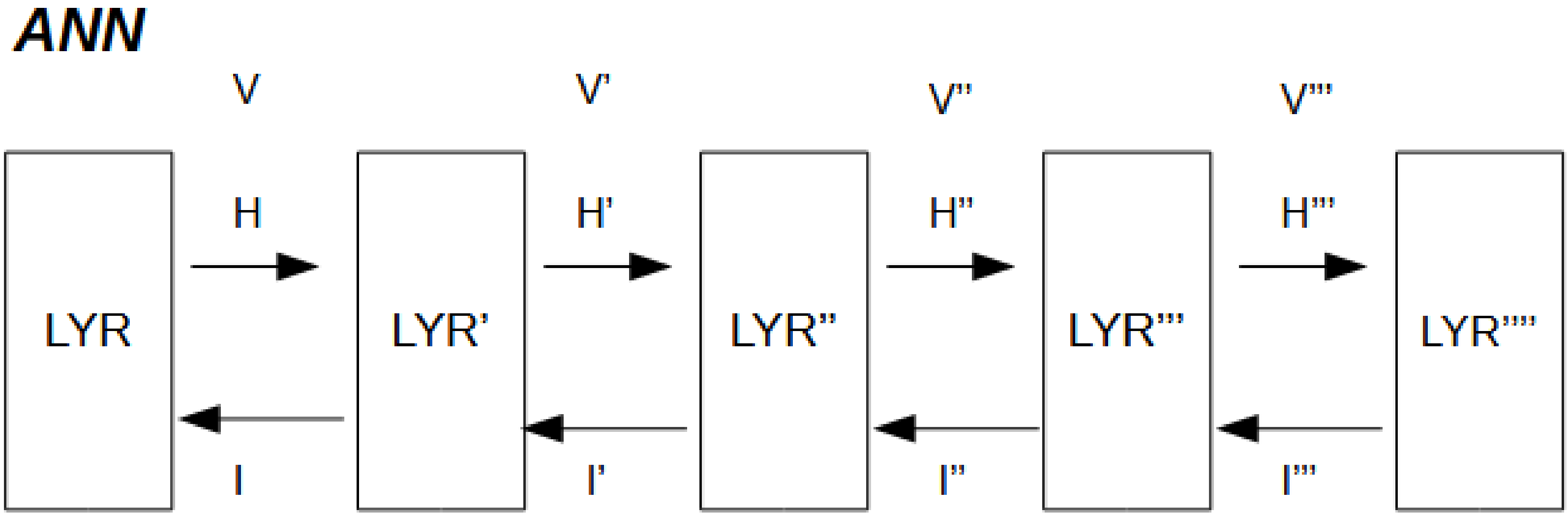
- Artificial Neural Nets (ANN)
- Fundamental object is the artificial neuron
  - A receptor or a transmitter of signals, often an algorithm
- A signal between neurons (synapses) can be weighted
- Neurons are assembled in layers
    - The opening layer receives inputs
    - The closing layer provides an output
    - Middle layers are hidden
- A neuron can have many inputs
  - But usually just one output (which can branch into multiple neurons)

Neural Net as a Transactional Monad

- Each layer is a closed monoidal category
  - With tensor product, weights.
  - Within a 2-category, with lax symmetry,
    - relaxing isomorphism and associativity requirements
    - giving non-linearity
- Between each layer is a
  - Free functor, goal-directed
  - Underlying functor, enforcing back propagation (training)
- Such a construction is a 2-monad
  - Provided some transactional conditions are satisfied
  - This monad can be composed with some housekeeping conditions (distributive e.g. Kleisii-lift in Cartesian)

ANN: Composition of Monads IH

V = IH



Each layer LYR is a 2-category: arrows between arrows

Each 2-monad is a 3-category: (arrows between arrows) between (arrows between arrows)

2-Monads

- Exist within a 3-category
- May be strict, fully weak, lax, pseudo, colax
- Pseudo preserves the structure up to a specified isomorphism
- So more scope for flexibility in strictness of associativity, isomorphism rules

- Example of ANN with image recognition to show grounding in real-world for part of model

Example of Composed 2-Monad **ANN** in Action

- Typing of layers is not homogenous
  - Left-hand layer for input LYR is pixels
    - px =(image_id,pixels)
      - image_id is the identifier for an image
      - pixels = height ⊗ width ⟶ channel
      - height and width are the position of the pixel in the grid and channel is the colour of the pixel
  - Right-hand layer for output LYR”” contains
    - the results from the application of the net, comparing prediction with real-life

Example – middle layers

- The middle layers are structures representing the neural net
  - typed as neurons not as pixels or results.
  - objects are, as a minimum, n for neuron, wt for weight, and patt for pattern.
    - The weight is the weight assigned to the connection in the net
    - The pattern is the shape to which the original pixel might be matched, such as an edge for the first layer LYR′

Example – the Triangle

- The arrows are n → wt, wt → patt, with the composite n → patt, giving a triangle

$$
\begin{tikzcd}
 & \mathrm{wt} \arrow[dr] & \\
\mathrm{n} \arrow[ur] \arrow[rr] & & \mathrm{patt}
\end{tikzcd}
$$

Generally the more middle layers there are, the more fine-tuned the net can be made with the patterns in the middle layers becoming progressively more refined in detail from left to right. So the patterns might be edges in LYR′, corners and other joined shapes in LYR″, and more complex patterns such as animal shapes in LYR‴.

Example – complex substructures

- The recognition of patterns involves some summation and clustering through algorithms in practice, which adds substructure to the categories, involving monoidal constructions.
- So there are extensive substructures which require tensor products for operations on them

Example – the Free Functors

- All functors from left to right are free functors
- Functor H maps px for pixels in the input LYR to n --> patt as triangles of neurons, weights and patterns in LYR′, the first inner layer
- Functor H′ maps the triangles in LYR′ to those in LYR″, refining the image pattern matching
- Functor H″ maps the triangles in LYR″ to those in LYR‴, again refining the image pattern matching
- Functor H‴ maps the node-weight-pattern triangles in LYR‴ onto the statistical outputs of LYR⁗.
    - This involves a match against the real-world answer as part of the training process; also recorded will be various statistical parameters measuring discrepancies, called losses
- In the matching process. It is very unlikely that a neural net will perform optimally on its first run

Example – the Underlying Functors

- These discrepancies are used in the training process in backpropagation, involving the underlying functors from right to left.
- Backpropagation functors run from right to left, starting with I′′′ which maps from the results in the category LYR′′′′ to the category on the extreme right of the middle layers containing the neurons LYR′′′.
- These functors are algorithms which calculate the gradient of a loss function with respect to each weight and use an optimiser to adjust each weight.
- They may also be used to adjust any biases that have been introduced into the neural net model. Such functors work back through the layers moving left to the input layer LYR.

Example -- Training

- Backpropagation is a crucial part of the training process: after the net's performance has been measured against the real-world; it enables parameters to be adjusted and a further attempt made.
- A cycle of forward and back processes is then run until the performance is judged to be optimal.
- In the category theory model in Figure 18 it looks at first glance as if this is a cyclic system.
- This is not the case: the adjointness inherent in the diagram involves a snap on the final solution, much like Whitehead's prehension in his Process & Reality.
- The training process provides an empirical grounding through the cases technique.

Formulation of 3-pullback to represent relationship between brain **ANN** and physical aspects of body **MB** with respect to the nervous system **N**

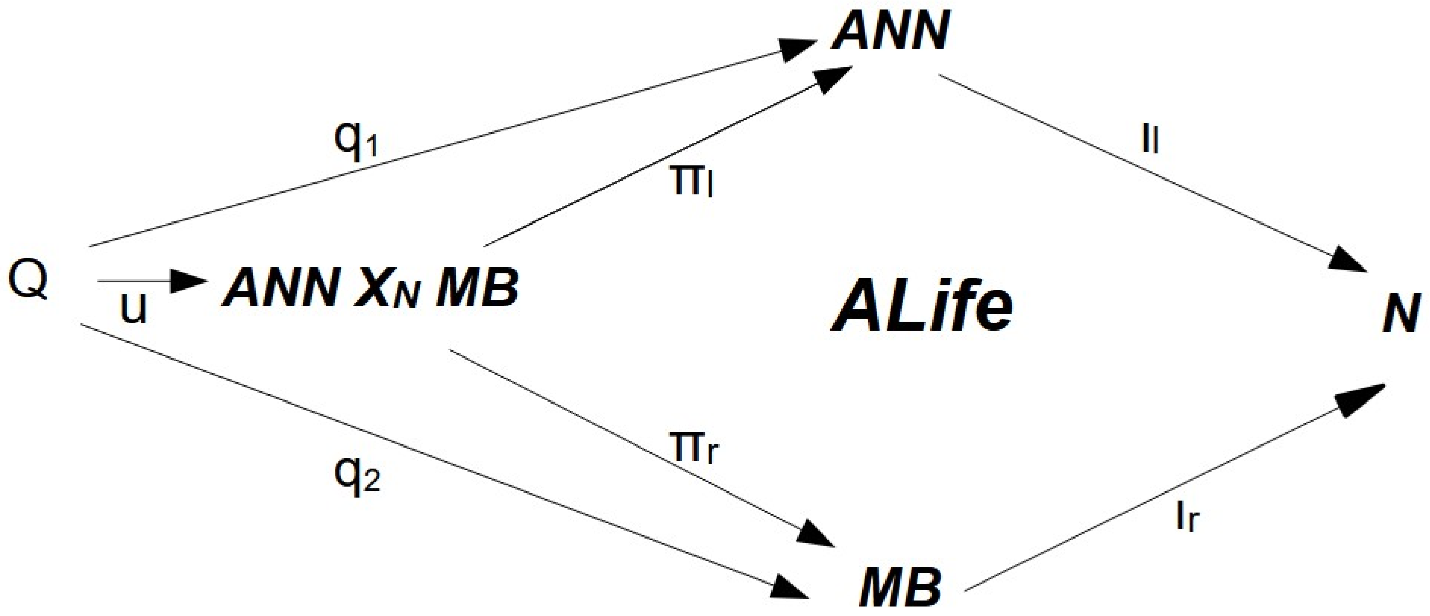
How do we connect a 1-Monad to a 2-Monad?

- Both **ANN** (2-monad, brain) and **MB** (1-monad, physicality) are arrows -- endofunctors
- Can we have an adjunction between them?
  - ANN $\dashv$ MB
  - Difficult to construct as different dimensions
    - 1-Monad is arrows between arrows
    - 2-Monad is (arrows between arrows) between (arrows between arrows)

Potential of Pullback

- Easier to relate the monads through a third entity
- The brain and physicality are in reality connected by the nervous system
  - Let us call this **N**
  - We will expand on **N** later
- This gives the relationship **ANN $\mathbf{X}_{\mathbf{N}}$ MB**
- This is a 3-pullback (a 3-category) allowing the expression of both the 1-monad and the 2-monad
    - (arrows between arrows) between (arrows between arrows)

3-Pullback for **ANN** over **MB** in context of **N**



Elaboration of 3-pullback as extension of work

Expansion of **N**: the Nervous System

- The Nervous System is generally considered to include the following:
  - The Central Nervous System (CNS, brain, spinal chord)
  - The Peripheral Nervous System (PNS, connections to the body)
  - Nerves are considered as bundles of fibres, another name for pullback logic
- Much of the nervous system can be considered as involving digital signals (0 or 1)
  - But the synapses (connections between neurons) and signals from sensory receptors are analogue (weighted by a real number). Examples of the latter:
    - Mechanoreceptors – skin, deformation, touch
    - Thermoreceptors – skin, hot or cold
    - Photoreceptors – eyes, light
    - Chemoreceptors – taste and smell detectors, oxygen and blood issues
    - Nociceptors – pain detection through tissue damage

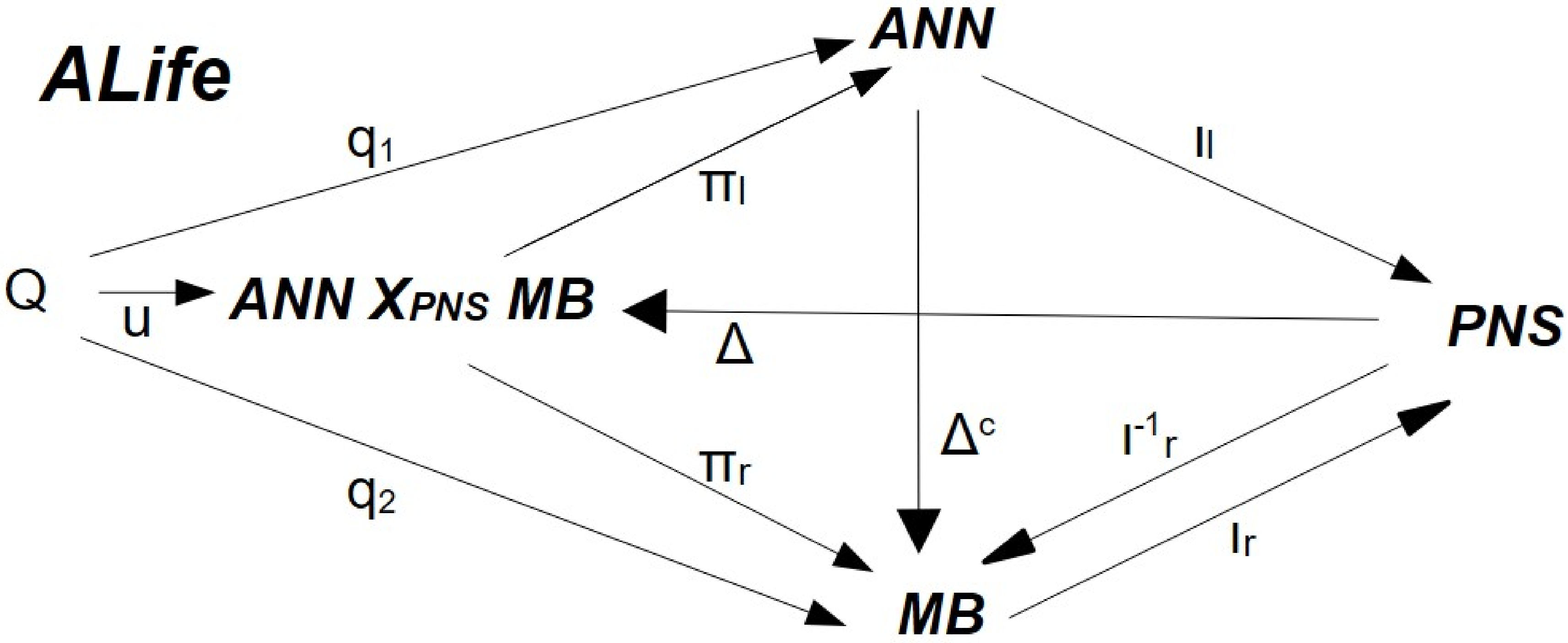
Refinement of our definitions

- ***ANN*** attempts to emulate the Central Nervous System (CNS)
  - Brain plus Spinal Chord
  - Highly reductionist attempt so not re-labelling
- ***MB*** is the physicality
  - The physical processes
- ***N*** is precisely the Peripheral Nervous System (PNS)
  - The connection between the CNS and the physical processes
  - So re-label ***PNS***

Afferent/Efferent Nerves

- Nerves have direction:
  - Afferent – inwards from ***PNS*** from sensors/physical extremities to the product ***ANN*** $\boldsymbol{X_{PNS}}$ ***MB***
  - Efferent – outwards from ***ANN*** to ***PNS*** to ***MB*** to stimulate motor actions
- Although we may not have an LCCC with the adjointness
  - Exists $\dashv$ Diagonal (Δ) $\dashv$ Universal
- There is always a diagonal in a pullback because of its universality:
  - Δ is the arrow from the third body (the independent variable) ***PNS*** to the product ***ANN*** $\boldsymbol{X_{PNS}}$ ***MB*** (the dependent variable)
  - Could be considered as the afferent nerves
- There is always also a counter-diagonal:
  - Δ^c is the arrow from the ***ANN*** to ***MB*** directly with composition through ***PNS***
  - Could be considered as the efferent nerves

3-Pullback for **ANN** over **MB** in context of **PNS** with additional arrows



Δ is afferent nerves; Δ^c is efferent nerves = $I^{-1}_r \circ I_l$

Nature of PNS

- ***PNS*** is a process
  - Can be emulated through a monad
  - The afferent nerves require quantitative weighting from the sensors
  - This weighting can be done through monoidal categories with tensor products
- So ***PNS*** is a 2-monad like ***ANN***
- So the 3-Pullback is of a 2-monad over a 1-monad in the context of a 2-monad

To conclude 1

- Category theory has developed from its simple Cartesian basis with natural transformations and adjointness
- n-categories provide more sophisticated mappings and relaxation of rules
- Monoidal categories facilitate tensor products
  - requiring 2-categories for the extra level
- Monads emulate processes, in particular as a pair of adjoint functors
  - 1-monad in a 2-category, 2-monad in a 3-category
- 3-categories begin to match the complexity of neural systems
- Relating the complex monads for the Nervous System appears to be best done through a 3-pullback

To conclude 2

- The 3-pullback diagram represents weak AI as it is based on Artificial Neural Nets (ANN)
- We cannot label ANN as CNS as there is no attempt to handle the Central Nervous System in our formalism
- To achieve Strong AI, ANN must be replaced by a formalisation of the Central Nervous System
- This is a major undertaking