Musical Performance: a Composition of Monads

Nick Rossiter & Michael Heather
Visiting Fellow
Computing Science and Digital Technologies
Northumbria University

UNILOG'2018
Vichy, France
Workshop on Music and Logic
25 June 2018

Acknowledgements

- Members of the Royal Northern Sinfonia,The Sage, Gateshead, UK, in particular:
  - Alexandra Raikhlina, Sub-principal 1st violin

Outline of Presentation 1

- Taking on the challenge of a testing application for the Cartesian monad (categorical) approach to universal design
  - Monad = process, operating on a topos
  - Topos = structure, Cartesian (product)
- Look at previous work using category theory with music

Outline of Presentation 2

- The need for natural application
  - Music is a composition of notes, with rules
  - Category theory is a composition of arrows, with rules

  Use top levels of category theory
    - With closure over three levels
  - Maximise expressiveness in data structuring
    - With topos and recursive intension/extension layers
  - Capture the process of performing music
    - With monad, operating within the topos

Earlier Work in Music with Category Theory and Nets 1

- Henry Klumpenhouwer/David Lewin (1991-2002)
  - K-nets and L-nets
  - Transformations from one pitch class to another
  - Graphical technique, using isographs
  - Classical harmony based on Z_{12}

Earlier Work in Music with Category Theory and Nets 2

- Guerino Mazzola (2002)
  - The Topos of Music
  - Develops functorial denotators, based on K-nets
    - Digraphs, graphical structures with edges from one node (pitch class) to another
    - Digraphs permit loops
  - Provides detailed exposition of theory of music
    - Useful for underlying musical structures
  - Rather disjoint treatment of title

Earlier Work in Music with Category Theory and Nets 3

- Guerino Mazzola and Moreno Andreatta (2006)
  - Develop denotators idea
    - Again pitch classes initially based on Z_{12}
    - Vertices are pitch classes; edges are transpositions
    - Digraph is constructed
    - Extend pitch classes with a 4-tuple for articulation: <onset, pitch, loudness, duration>
    - Use powerobjects to represent chords
    - Paths are maintained through the music
  - Work appears to be based on the Eilenberg-Moore category: the pullback of the category of presheaves on the Kleisli category along the Yoneda embedding (not cited as such)

Earlier Work in Music with Category Theory and Nets 4

- Alexandre Popoff, Carlos Agon, Moreno Andreatta, Andrée Ehresmann (2016a)
  - Developed generalised PK-nets (poly-K)
    - Giving more flexibility in structure of nodes
      - Variable cardinalities
      - Labelling flexible for different genre
    - Defining a natural transformation between functors to achieve the flexibility
- Shortly after (2016b), they introduced the REL category for relationships within nodes, replacing SET

Feelings on Earlier Work

- Sound advance in basic musical structures
- But some of the work appears to be categorification
  - Direct 1:1 translation from set (graph) theory
- And there may be more natural methods for
  - Composition
  - Data structuring with topos
  - Process or communication
- That on denotators in 2006 (Guerino Mazzola and Moreno Andreatta) with apparent use of the Eilenberg-Moore category comes closest to the ideas presented here
- And the PK-nets or denotators could be used as a representation at an underlying level for the score, helping higher-level workers

The Aim is a Topos – Structural Data-type

- Based on Locally Cartesian Closed Category (LCCC) [Descartes]
  - relationships within a product (pullbacks, limits)
  - connectivity (exponentials)
  - internal logic (λ-calculus)
  - identity (from the limit)
  - interchangeability of levels (object to category-object)
  - hyperdoctrine (adjointness between quantifiers and the diagonal)

The Aim is a Topos – Structural Data-type 2

- If we add:
  - definition of relationships within a coproduct (colimits)
  - internal intuitionistic logic (Heyting)
  - subobject classifier (query)
  - reflective subtopos viewpoint (query closure)
- We get a Topos [Aristotle]

A Topos for Music

- Music is viewed as a communication of some manuscript by communicators
- The topos is relatively static (compared to the monad) but being arrow-based can readily handle change.
- Manuscript comprises scores and other intentions of composers and writers
  - Includes musical notation (typeset, handwritten or digital) or more spontaneous formats
- Communicators comprise performers and other aspects of performance
  - Includes an orchestra, group, recording company

Intension/Extension [Aristotle]

- Arguably the most important feature in music
  - Terms come from philosophy
- In mathematics/computing science:
  - The intension is the type, the extension is the collection of instances that satisfy the type
- It's not as simple though as a hierarchy of types
  - There remains a philosophical dimension

Universe of Discourse

- The Universe contains everything
- The Universe of Discourse (UoD) is that section of the Universe of interest to our application
- By the laws of physics we cannot isolate any part of the Universe but we can identify a section for our work
- In this case
  - The intension is the Universe
  - The extension is the world of music (UoD)

A Score is far from fixed

- A musical manuscript has both
  - Intensional properties
    - as a type for how the work is to be performed
    - according to the composer
  - Extensional properties
    - as instances of the manuscript
    - according to variants in
      - Publication (composer initiated, developments after composer's death)
      - Rehearsal (conductor initiated)
      - Performance
        - No two performances are ever the same

A Manuscript is both extensional and intensional

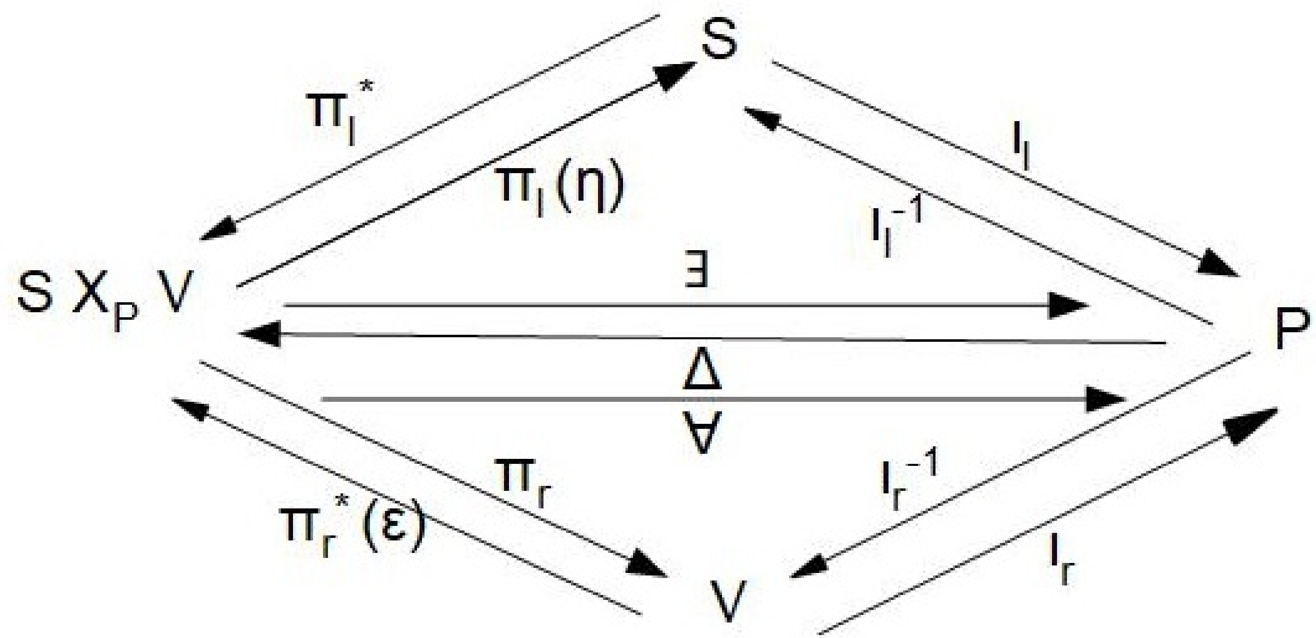
- A musical manuscript is extensional to the Universe of Discourse of Music
  - One of the objects in this universe
- But intensional to the manuscript, its variants and their performances
  - Defining the underlying objects
- So elaborate intension/extension hierarchies can be constructed
  - Where there is a genuine semantic change
  - Category theory suggests four levels are adequate

Pullbacks for Relationships

- In category theory relationships can be represented by:
  - Products (unqualified X)
  - Pullbacks (qualified X)
  - Union (unqualified +)
  - Pushouts (qualified +)
- The pullback for the Manuscript/ Variant/ Performance relationship follows

Relationship of Score by Variant in Context of Performance (S X_P V)

- Pullback – Locally Cartesian Closed Category



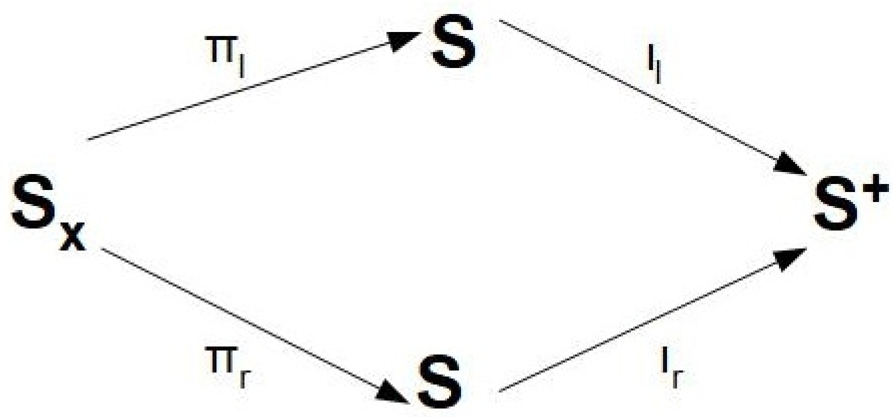
S is category for Score, V for Variant, P for Performance

Realising the Extensional Part

- Preceding diagram appears to be the intensional structure
  - The definition (or type structure)
- There is also the extension
  - The instances (conforming to the type structure)
- The diagram actually does include the extension as well, within the category-objects of the pullback $S \: X_P \: V$, S, P, V

Intension/Extension Relationship for Category-object S

- Type/Instance as Dolittle Diagram



As π_l is monic, then so is I_r : diagram is both a pullback and a pushout with limit, product, colimit, coproduct

It's an adhesive category, also known as a pulation square

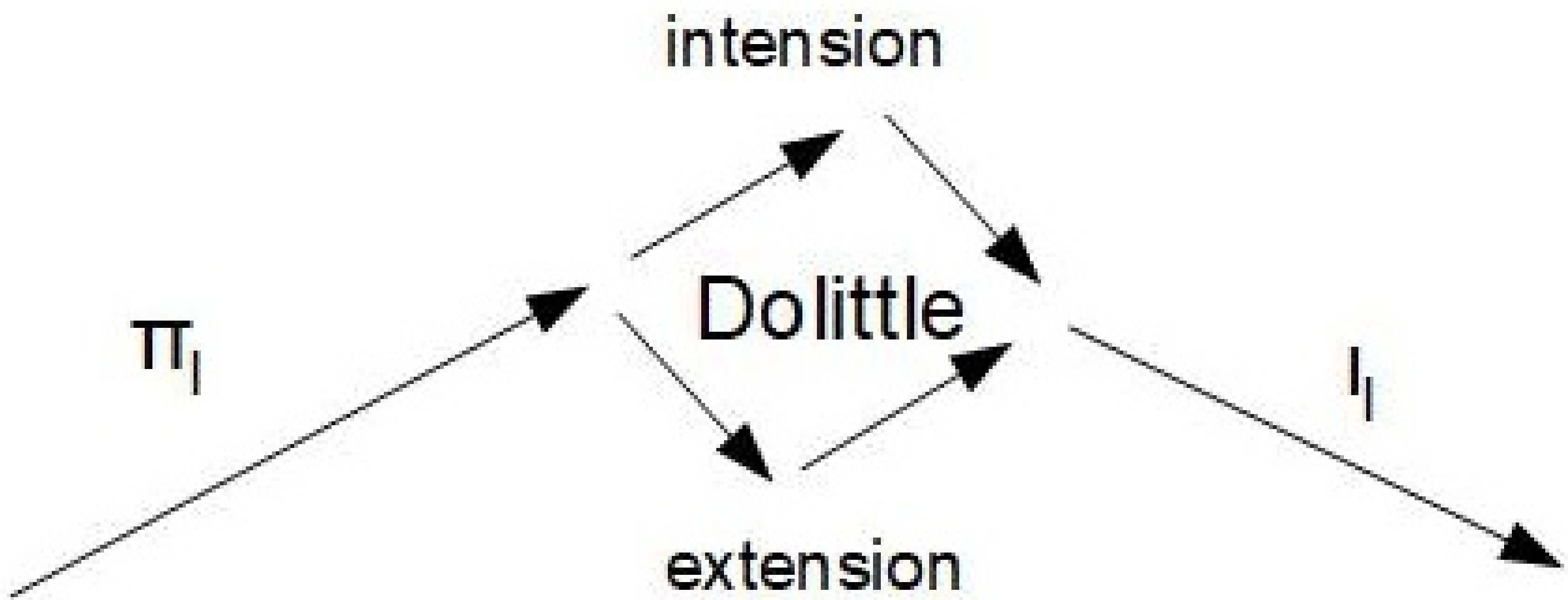
S is score; top S is intension (type); bottom S is extension (set-values); S_X is limit (type X value pairs); S^+ is colimit (type + value pairs)
The extension for the score will contain the notes, perhaps as digraphs

Intension/Extension and the Topos

- Every node (category-object) in our pullback relationships will contain such an:
  - Intension part
  - Extension part
- in a Dolittle structure
  - also known as Pulation square, Adhesive category
- Adhesive categories are readily embedded into a topos Є

Category-object Expanded

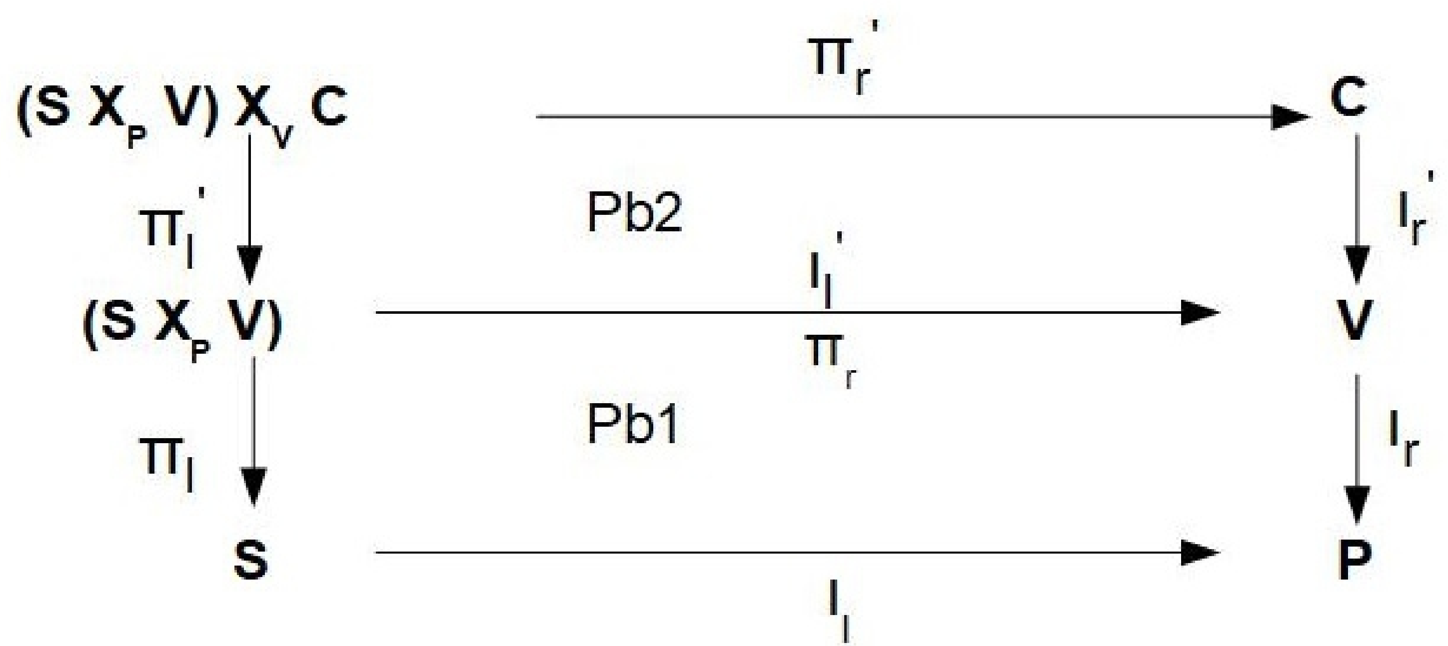
- Category-object S expanded in pullback S X_P V



Data Structuring with Pullbacks

- In real-world, nodes contain more structure than shown
- Also the real-world is more complex than one pullback.
- Need to build more complex structures than that shown. Could:
  - Expand category-objects with further levels
  - Paste pullbacks together
    - Pursued in information systems with satisfactory results
- This is still an experimental area

Example of Pasted Pullback



Have 3 pullbacks: Pb2 X Pb1; Pb2; Pb1

C is category-object for Composer

Overall relationship is of Score with Variant and Composer in context of Performance

Process within the Topos

- Philosophy
  - Metaphysics (Whitehead *Process and Reality* 1929)
  - All is flux [Heraclitus]
- Transaction (Universe or information system)
- Activity
  - Can be very complex but the whole is viewed as atomic – binary outcome – succeed or fail
  - Before and after states must be consistent in terms of rules
  - Intermediate results are not revealed to others
  - Results persist after end

Promising Technique – Monad

- Philosophy of Leibniz
  - Elementary 'substance' whose interior cannot be examined (encapsulation)
- The monad is used in pure mathematics for representing process
  - Has 3 'cycles' of iteration to give consistency

Monad in Functional Programming

- The monad is used to formulate the process in an abstract data-type
- In the Haskell language the monad is a first-class construction
  - Haskell B Curry transformed functions through currying in the λ-calculus
  - The Blockchain transaction system for Bitcoin and more recently other finance houses uses monads via Haskell
  - Reason quoted: it is simple and clean technique
  - Shortage of Haskell programmers has encouraged the use of the Python language

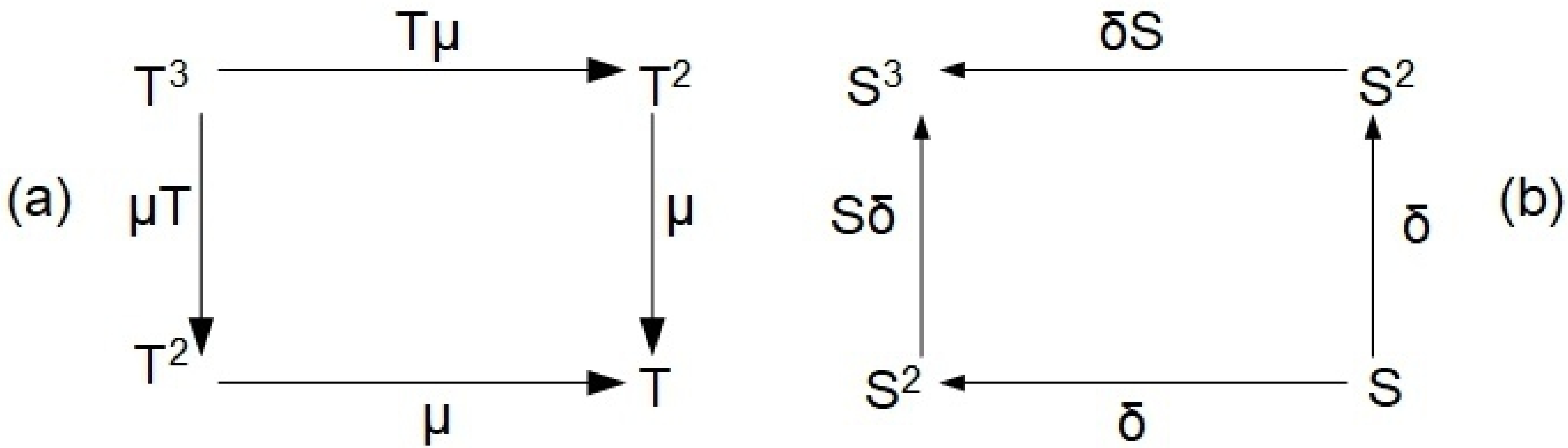
Monad can be based on an adjunction

- The transaction involves GF, a pair of adjoint functors F -| G
  - F: X → Y
  - G: Y → X
- GF is an endofunctor as category X is both source and target
- So T is GF (for monad)
- And S is FG (for comonad)

Monad/Comonad Overview

- Functionality for free functor T, underlying functor S
  - Monad
    - $T^3 \rightarrow T^2 \rightarrow T$ (multiplication)
    - 3 'cycles' of T (GFGFGF)
  - Comonad (dual of monad)
    - $S \rightarrow S^2 \rightarrow S^3$ (comultiplication)
    - 3 'cycles' of S (FGFGFG)
- Objects
  - An endofunctor on category Є (the topos)
- Note this multiple performance matches our transaction approach, outlined earlier, with GF performed 3 times

Monad/Comonad Diagrams

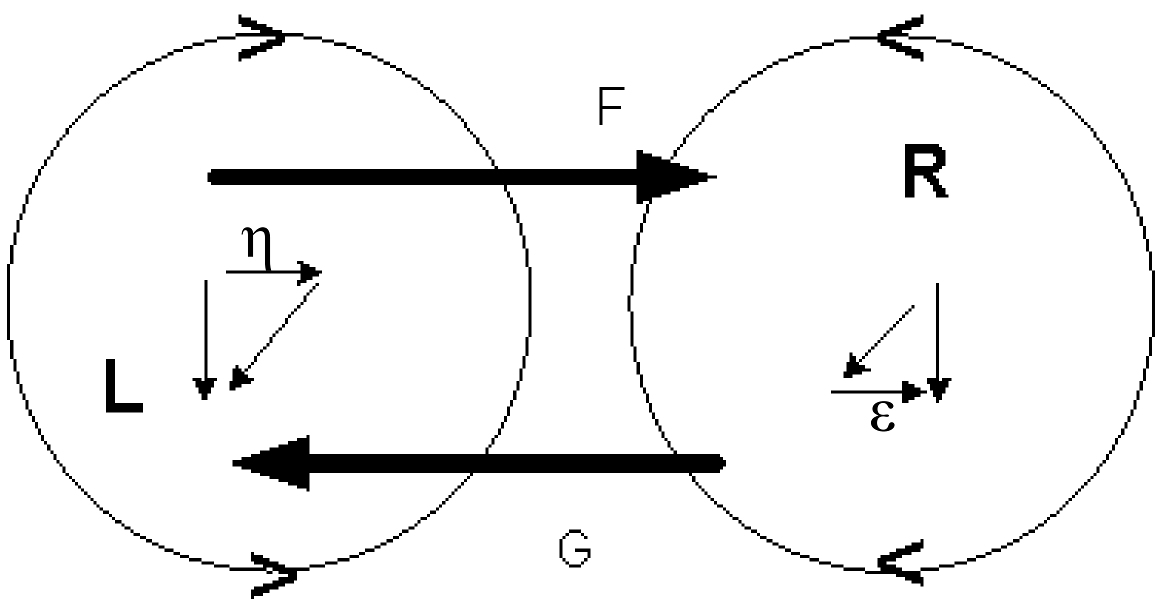


(a) the monad construction $T^3 \rightarrow T^2 \rightarrow T$ where T = GF, multiplication = μ

(b) the comonad construction $S \rightarrow S^2 \rightarrow S^3$ where S = FG, comultiplication = δ

A 'Cycle' representing adjointness

- One ‘cycle’ for monad T (GF)
  - Assessing unit η in L and counit ε in R to ensure overall consistency
  - One or more 'Cycles' is performed simultaneously (a snap, not each cycle in turn).



F -| G

$\eta: 1_L \rightarrow GF(L)$ $\varepsilon: FG(R) \rightarrow 1_R$

Failure in Expected Adjointness

- Means transaction has failed
- Communication is suspended
- Restart is necessary at some convenient point (Rollback)
- In music need to distinguish between a wrong note and differences in expression:
  - Intonation is the rules, on violin with left hand (left exact)
  - Articulation is the expression, on violin with right hand (right exact)
- Failure leads to revised adjointness

Operating within a Topos

- The monad operation is simple:
  - T: Є → Є
    - where T is the monad <GF, η, μ> in Є, the topos, with input and output types the same
- The extension (data values) will vary but the intension (definition of type) remains the same
- Closure is achieved as the type is preserved

Process in Musical Performance

- The topos Є created earlier contains
  - The intension/extension in the categories S (for Score, with musical notation), V (for Variant), $S\ X_P\ V$ (for their qualified product), P (for performance for the actual musical event)
- A single monad/comonad action (of 3 cycles T^3) will take the music forward one unit of performance (phrase or bar), say one step

Process in Musical Performance 2

- Moving from one barline to another is determined uniquely by the adjunction F -| G
  - F is the free functor (looking forward, creative/expressive)
  - G is the underlying functor (looking back, enforcing the rules, qualia)

Process in Musical Performance 3

- If adjointness holds over the 3 cycles
  - Then η the unit of adjunction measures the creativity of the step going forward (rhetoric)
  - And ε the counit of adjunction measures the qualia of the step looking back (dialectic)
- If expected adjointness does not hold over the 3 cycles
  - Then integrity has been lost and resynchronization is necessary with revised adjointness

Comparison with Earlier work

- Our end-product bears some similarity to the denotators described earlier but
  - uses the topos and the monad to represent process at a conceptual level
  - is more suited for further development of the topic
  - encouraging discussion with musicians.

Experience

- Performers do comment that playing is an intensive experience:
  - at the same time both looking back as to what you have played and anticipating what is to come.
- Such experience is captured by the monad/comonad structure with its forward/backward nature and inherent adjointness

Monad/Comonad Direction

- Overall the monad looks backwards
  - $T^3 \to T^2 \to T$
- and its comonad forwards
  - $S \to S^2 \to S^3$
- in their three cycles.
- However, the situation is more subtle than this:
  - in each cycle the monad looks forwards (F) and then backwards (G)
  - its comonad looks backwards (G) and then forwards (F)

Orderly Communication

- The duality of the monad/comonad represents communication in an orderly manner within initially defined colimits and adjointness.

Composition

- A musical work is referred to as a composition.
- It is indeed a composition of steps
  - With the output from one step becoming the input to the next step
- The order is fixed in advance
- Composition is an inherent feature of category theory
- With one monad execution as a single step, it is necessary to compose monads to perform a full work

Therefore composability is the Key

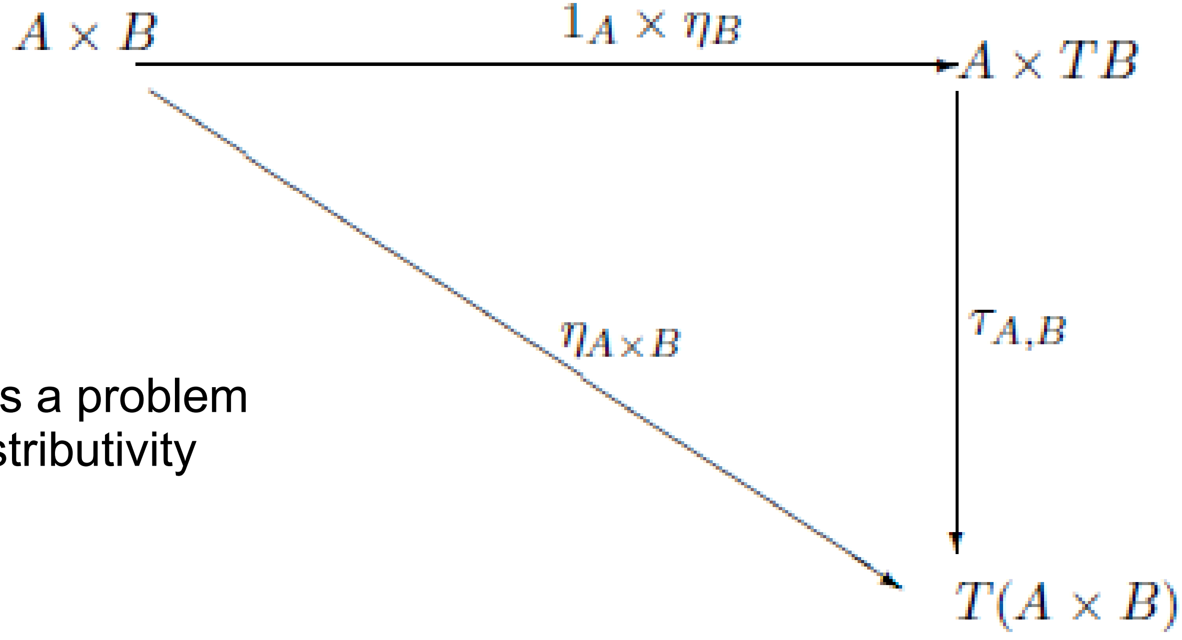
- Compose many monads together to give the power of adjointness over a whole wide-ranging application
- In banking (Bitcoin) the reliability obtained from composing processes over a wide-range of machines (distributed data recovery) justifies the move to Category Theory
- There is a problem though in EML (Eilenberg/ Mac Lane) Category Theory:
  - Monads do not compose naturally

Haskell and Monads

- Kleisli Category of a Monad
  - Transforms a monad into a monadic form more suitable for implementation in a functional language
    - Used in Haskell rather than the pure mathematics form of Mac Lane
- Strengthens the monad for composability
  - As in the Cartesian Monad, with products
- A practical application of the pure maths has exposed problems in the maths
- Solution has come from another pure mathematician Kleisli

Kleisli Lift

- Define a natural transformation:
  - $\tau_{A,B}$: A X TB → T (A X B) where A,B are objects in X and T is the monad such that the following diagram commutes



There is a problem
with distributivity
In EML

Cartesian Monads in Music

- Take each barline, or some other time signature, as a unit of process
  - Such a barline will be Cartesian, representing the potentially complex physics of the music
    - Combinations of notes, including chords
    - Or powerobjects as in the denotators approach
- Therefore Cartesian Monads as strengthened by the Kleisli Lift are essential for composition purposes

Summary of Progress/Look forward

- Topos has been established as data-type of choice
- Monad shows potential for processing the topos
- There is no assumption of any particular musical genre.
- Such a categorial framework could be implemented in the functional programming language Haskell
  - Basic physical music structures have been implemented in Haskell (Paul Hudak)

References

- Hudak, Paul, The Haskell School of Music - From Signals to Symphonies - Yale University, Department of Computer Science, Version 2.4 353 pp, February 22 (2012).
- Klumpenhouwer, Henry: A Generalized Model of Voice-Leading for Atonal Music. Ph.D. Thesis, Harvard University (1991)
- Lewin, David: Thoughts on Klumpenhouwer Networks and Perle-Lansky Cycles. Music Theory Spectrum 24.2: 196-230 (2002).
- Mazzola, Guerino et al.: The Topos of Music -Geometric Logic of Concepts, Theory, and Performance. Birkhäuser, Basel, 1335pp (2002).
- Mazzola, Guerino and Andreatta, Moreno, From a Categorical Point of View: K-Nets as Limit Denotators, Perspectives of New Music Vol. 44, No. 2, pp. 88-113 (2006).
- Popoff, Alexandre, Agon, Carlos, Andreatta, Moreno and Ehresmann, Andrée, From K-Nets to PK-Nets: A Categorical Approach, Perspectives of New Music, Vol. 54, No. 2, pp. 5-63 (2016).
- Popoff, Alexandre, Andreatta, Moreno and Ehresmann, Andrée, Relational PK-Nets for Transformational Music Analysis, arXiv:1611.02249, 19pp, submitted 2016.
- Rossiter, Nick, Michael Heather & Michael Brockway, Monadic Design for Universal Systems, ANPA 37-38, Anton L. Vrba (ed.), St John's College, Rowlands Castle, Hampshire, UK, 8-12 August 2016, 369-399. (2018). http://nickrossiter.org.uk/process/Rossiter-ANPA-PROC-37-38-2%20final.pdf
- Whitehead, A N, Process and Reality: An Essay in Cosmology, Macmillan Publishers, New York, 1929, corrected edition by D.R. Griffin & D.W. Sherburne, Free Press, New York, (1978). https://monoskop.org/File:Whitehead Alfred North Process and Reality corr ed 1978.pdf