Category theory, higher dimensional algebra, groupoid atlases

Prospective descriptive tools in theoretical neuroscience

Ronnie Brown

Asklöster

July 25, 2009

Category theory, higher dimensional algebra, groupoid atlases

Prospective descriptive tools in theoretical neuroscience

Ronnie Brown

Asklöster

July 25, 2009

A few possibilities!

Challenge: The cricket cercal system

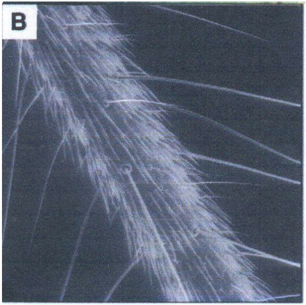
Challenge: The cricket cercal system

G.A. Jacobs, J.P. Miller, Z.Aldworth, 'Computational mechanosensory processing in the cricket', J. Exp. Biol., 211 (2008) 1819-1828.

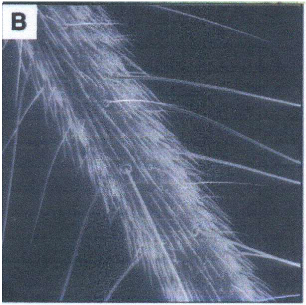

A

A. Acheta domestica. The cerci are the two antenna-like structures, covered with fine hairs, extending from the rear of the abdomen. This is a female: the ovipositor can be distinguished between the two cerci.

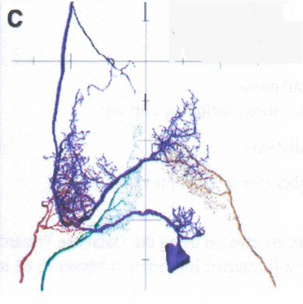
A. Acheta domestica. The cerci are the two antenna-like structures, covered with fine hairs, extending from the rear of the abdomen. This is a female: the ovipositor can be distinguished between the two cerci.

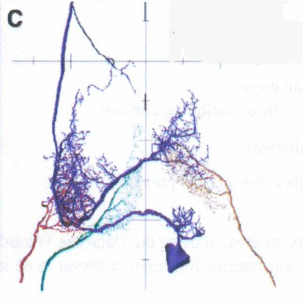


A. Acheta domestica. The cerci are the two antenna-like structures, covered with fine hairs, extending from the rear of the abdomen. This is a female: the ovipositor can be distinguished between the two cerci.

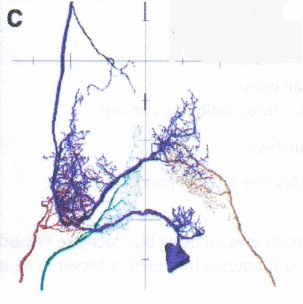


B. Scanning electron microscope close-up of a segment of the cercus. The cercus is approximately 1 cm in length.

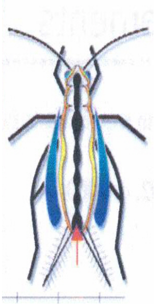

C

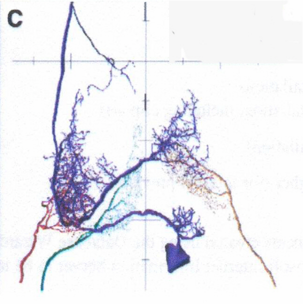


C. Computer reconstructions of a primary sensory Interneuron (blue) and three primary sensory afferents (red, light blue and brown) in their correct anatomical relationships. Scale: 40 μm between tick marks on the scale bars.

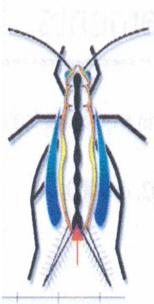


C. Computer reconstructions of a primary sensory Interneuron (blue) and three primary sensory afferents (red, light blue and brown) in their correct anatomical relationships. Scale: 40 μm between tick marks on the scale bars.





C. Computer reconstructions of a primary sensory Interneuron (blue) and three primary sensory afferents (red, light blue and brown) in their correct anatomical relationships. Scale: 40 μm between tick marks on the scale bars.



Cartoon of a cut-away view of the cricket nervous system. The terminal abdominal ganglion, where the sensory neurons and interneurons are located, is indicated with a red arrow.

The computations these interneurons appear to carry out include (at least) the following operations:

The computations these interneurons appear to carry out include (at least) the following operations:
(a) noise reduction through signal averaging across many sensory afferents,

The computations these interneurons appear to carry out include (at least) the following operations:
(a) noise reduction through signal averaging across many sensory afferents,
(b) extremely efficient re-encoding of the direction of air current stimuli, via a huge dimensional reduction of the activities of the 1500 sensory afferents down to a four-interneuron ortho-normalized code

The computations these interneurons appear to carry out include (at least) the following operations:
(a) noise reduction through signal averaging across many sensory afferents,
(b) extremely efficient re-encoding of the direction of air current stimuli, via a huge dimensional reduction of the activities of the 1500 sensory afferents down to a four-interneuron ortho-normalized code
(c) coding of the spectral composition of dynamic air currents via the relative activity levels of different interneurons having different frequency sensitivity bands, and

(d) (still speculative) the representation of the curl of air currents via interneurons sensitive to vortices rather than linear air streams. The information available at this first-order sensory interneuron interface is extraordinarily good from an engineering perspective, in terms of the temporal and angular accuracy and precision.

(d) (still speculative) the representation of the curl of air currents via interneurons sensitive to vortices rather than linear air streams. The information available at this first-order sensory interneuron interface is extraordinarily good from an engineering perspective, in terms of the temporal and angular accuracy and precision.

Based on this generalist information represented at the first stage of processing, more complex processing operations (such as feature detection and target identification) are, presumably, computed at higher levels of the nervous system by more specialized cells and circuits.

(d) (still speculative) the representation of the curl of air currents via interneurons sensitive to vortices rather than linear air streams. The information available at this first-order sensory interneuron interface is extraordinarily good from an engineering perspective, in terms of the temporal and angular accuracy and precision.

Based on this generalist information represented at the first stage of processing, more complex processing operations (such as feature detection and target identification) are, presumably, computed at higher levels of the nervous system by more specialized cells and circuits.

Category theory:

Category theory:
an abstract setting for analogy and comparison

Category theory:
an abstract setting for analogy and comparison

Category theory adds to directed graphs the notion of composition.

Category theory:
an abstract setting for analogy and comparison

Category theory adds to directed graphs the notion of composition.

Notable as a partial algebraic system whose composition is defined under geometric conditions.

Category theory:
an abstract setting for analogy and comparison

Category theory adds to directed graphs the notion of composition.

Notable as a partial algebraic system whose composition is defined under geometric conditions.

Central concepts: limit and colimit.

Category theory: an abstract setting for analogy and comparison

Category theory adds to directed graphs the notion of composition.

Notable as a partial algebraic system whose composition is defined under geometric conditions.

Central concepts: limit and colimit.

Colimits give a general concept of gluing, or integration, of objects.

Category theory:
an abstract setting for analogy and comparison

Category theory adds to directed graphs the notion of composition.

Notable as a partial algebraic system whose composition is defined under geometric conditions.

Central concepts: limit and colimit.

Colimits give a general concept of gluing, or integration, of objects.

This notion allows analogies between wildly different constructions across wide fields of mathematics.

Category theory: an abstract setting for analogy and comparison

Category theory adds to directed graphs the notion of composition.

Notable as a partial algebraic system whose composition is defined under geometric conditions.

Central concepts: limit and colimit.

Colimits give a general concept of gluing, or integration, of objects.

This notion allows analogies between wildly different constructions across wide fields of mathematics. Such notions have made a great contribution in the second half of the 20th century to the unification of mathematics.

Category theory:
an abstract setting for analogy and comparison

Category theory adds to directed graphs the notion of composition.

Notable as a partial algebraic system whose composition is defined under geometric conditions.

Central concepts: limit and colimit.

Colimits give a general concept of gluing, or integration, of objects.

This notion allows analogies between wildly different constructions across wide fields of mathematics. Such notions have made a great contribution in the second half of the 20th century to the unification of mathematics.

Colimit is defined by its relation to, and

Category theory:
an abstract setting for analogy and comparison

Category theory adds to directed graphs the notion of composition.

Notable as a partial algebraic system whose composition is defined under geometric conditions.

Central concepts: limit and colimit.

Colimits give a general concept of gluing, or integration, of objects.

This notion allows analogies between wildly different constructions across wide fields of mathematics. Such notions have made a great contribution in the second half of the 20th century to the unification of mathematics.

Colimit is defined by its relation to, and communication with, all other objects.

Intuitions:

The object $\mathrm{colim}(D)$ is 'put together' from the constituent diagram D by means of the colimit cocone.

Intuitions:

The object colim(D) is ‘put together’ from the constituent diagram D by means of the colimit cocone. From beyond (or above our diagrams) D, an object C ‘sees’ the diagram D ‘mediated’ through its colimit, i.e. if C tries to interact with the whole of D, it has to do so via colim(D).

Intuitions:

The object colim(D) is 'put together' from the constituent diagram D by means of the colimit cocone. From beyond (or above our diagrams) D, an object C 'sees' the diagram D 'mediated' through its colimit, i.e. if C tries to interact with the whole of D, it has to do so via colim(D). The colimit cocone is a kind of program:

Intuitions:

The object colim(D) is 'put together' from the constituent diagram D by means of the colimit cocone. From beyond (or above our diagrams) D, an object C 'sees' the diagram D 'mediated' through its colimit, i.e. if C tries to interact with the whole of D, it has to do so via colim(D). The colimit cocone is a kind of program: given any cocone on D with vertex C, the output will be a morphism

$$\Phi : \mathrm{colim}(D) \to C$$

constructed from the other data.

Intuitions:

The object colim(D) is 'put together' from the constituent diagram D by means of the colimit cocone. From beyond (or above our diagrams) D, an object C 'sees' the diagram D 'mediated' through its colimit, i.e. if C tries to interact with the whole of D, it has to do so via colim(D). The colimit cocone is a kind of program: given any cocone on D with vertex C, the output will be a morphism

$$\Phi : \mathrm{colim}(D) \to C$$

constructed from the other data. How is this done?

Intuitions:

The object colim(D) is 'put together' from the constituent diagram D by means of the colimit cocone. From beyond (or above our diagrams) D, an object C 'sees' the diagram D 'mediated' through its colimit, i.e. if C tries to interact with the whole of D, it has to do so via colim(D). The colimit cocone is a kind of program: given any cocone on D with vertex C, the output will be a morphism

$$\Phi : \mathrm{colim}(D) \to C$$

constructed from the other data. How is this done?

Email analogy

Intuitions:

The object colim(D) is 'put together' from the constituent diagram D by means of the colimit cocone. From beyond (or above our diagrams) D, an object C 'sees' the diagram D 'mediated' through its colimit, i.e. if C tries to interact with the whole of D, it has to do so via colim(D). The colimit cocone is a kind of program: given any cocone on D with vertex C, the output will be a morphism

$$\Phi : \mathrm{colim}(D) \to C$$

constructed from the other data. How is this done?

Email analogy

You want to send an email Φ of a document D to a receiver C.

Intuitions:

The object $\mathrm{colim}(D)$ is 'put together' from the constituent diagram D by means of the colimit cocone. From beyond (or above our diagrams) D, an object C 'sees' the diagram D 'mediated' through its colimit, i.e. if C tries to interact with the whole of D, it has to do so via $\mathrm{colim}(D)$. The colimit cocone is a kind of program: given any cocone on D with vertex C, the output will be a morphism

$$\Phi : \mathrm{colim}(D) \to C$$

constructed from the other data. How is this done?

Email analogy

You want to send an email Φ of a document D to a receiver C. The document D is made up of lots of parts. The email programme

Intuitions:

The object $\mathrm{colim}(D)$ is 'put together' from the constituent diagram D by means of the colimit cocone. From beyond (or above our diagrams) D, an object C 'sees' the diagram D 'mediated' through its colimit, i.e. if C tries to interact with the whole of D, it has to do so via $\mathrm{colim}(D)$. The colimit cocone is a kind of program: given any cocone on D with vertex C, the output will be a morphism

$$\Phi : \mathrm{colim}(D) \to C$$

constructed from the other data. How is this done?

Email analogy

You want to send an email Φ of a document D to a receiver C. The document D is made up of lots of parts. The email programme splits D up in some way into pieces,

Intuitions:

The object $\text{colim}(D)$ is 'put together' from the constituent diagram D by means of the colimit cocone. From beyond (or above our diagrams) D, an object C 'sees' the diagram D 'mediated' through its colimit, i.e. if C tries to interact with the whole of D, it has to do so via $\text{colim}(D)$. The colimit cocone is a kind of program: given any cocone on D with vertex C, the output will be a morphism

$$\Phi : \text{colim}(D) \to C$$

constructed from the other data. How is this done?

Email analogy

You want to send an email Φ of a document D to a receiver C. The document D is made up of lots of parts. The email programme splits D up in some way into pieces, labels each piece at the beginning and end,

Intuitions:

The object colim(D) is 'put together' from the constituent diagram D by means of the colimit cocone. From beyond (or above our diagrams) D, an object C 'sees' the diagram D 'mediated' through its colimit, i.e. if C tries to interact with the whole of D, it has to do so via colim(D). The colimit cocone is a kind of program: given any cocone on D with vertex C, the output will be a morphism

$$\Phi : \mathrm{colim}(D) \to C$$

constructed from the other data. How is this done?

Email analogy

You want to send an email Φ of a document D to a receiver C. The document D is made up of lots of parts. The email programme splits D up in some way into pieces, labels each piece at the beginning and end, and sends these labelled pieces separately to C which combines them.

Intuitions:

The object colim(D) is ‘put together’ from the constituent diagram D by means of the colimit cocone. From beyond (or above our diagrams) D, an object C ‘sees’ the diagram D ‘mediated’ through its colimit, i.e. if C tries to interact with the whole of D, it has to do so via colim(D). The colimit cocone is a kind of program: given any cocone on D with vertex C, the output will be a morphism

$$\Phi : \mathrm{colim}(D) \to C$$

constructed from the other data. How is this done?

Email analogy

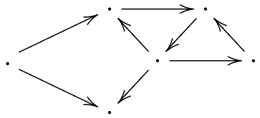
You want to send an email Φ of a document D to a receiver C. The document D is made up of lots of parts. The email programme splits D up in some way into pieces, labels each piece at the beginning and end, and sends these labelled pieces separately to C which combines them.
Also you want that the final received email is independent of all the choices that have been made.

Input data for a colimit:

Input data for a colimit:
a diagram D of arrows in a category

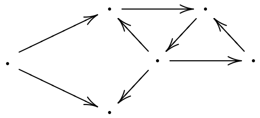
Input data for a colimit:
a diagram D of arrows in a category

$D =$

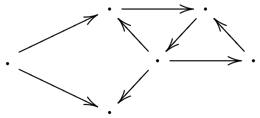


'Functional controls': cocone *with base D and vertex an object C.*

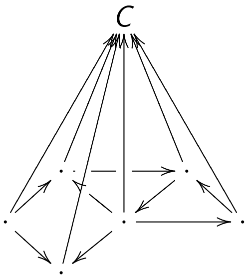
Input data for a colimit:
a diagram D of arrows in a category

$D =$ 

Input data for a colimit:
a diagram D of arrows in a category

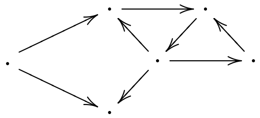
$D =$ 

'Functional controls': cocone *with base D and vertex an object C.*

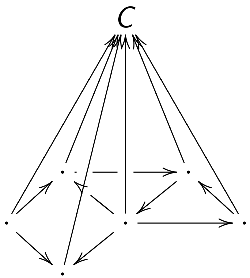


Input data for a colimit:
a diagram D of arrows in a category

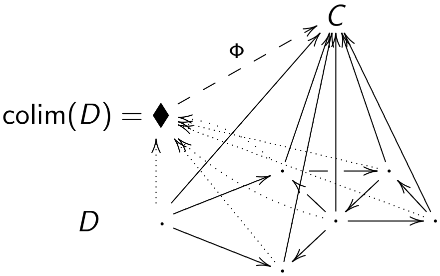
$D =$

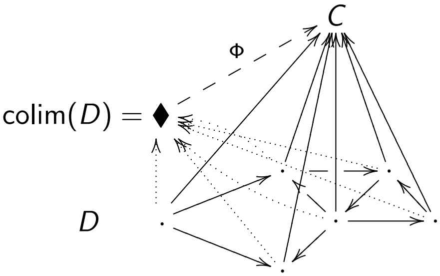


'Functional controls': cocone *with base D and vertex an object C.*

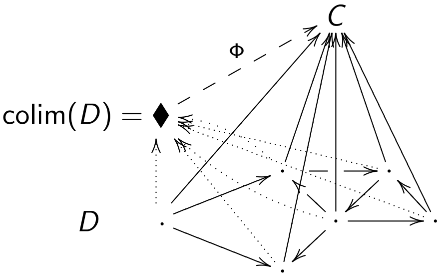


such that each of the triangular faces of this cocone is commutative.

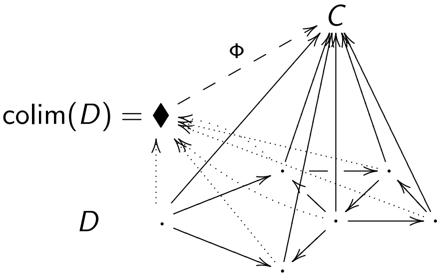

C
Φ
$\mathrm{colim}(D) = \blacklozenge$
D



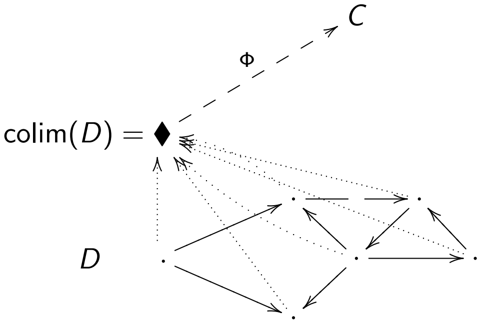
The output will be an object $\mathrm{colim}(D)$ in our category C defined by a special *colimit cocone*

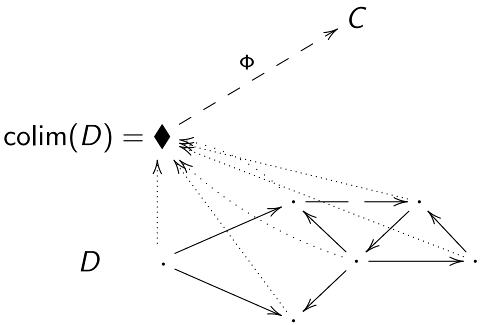


The output will be an object colim(D) in our category C defined by a special *colimit cocone* such that any cocone on D factors uniquely through the colimit cocone.

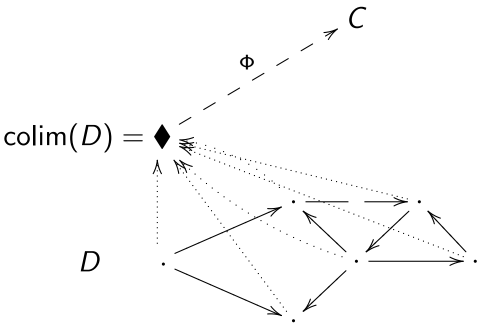


The output will be an object colim(D) in our category C defined by a special *colimit cocone* such that any cocone on D factors uniquely through the colimit cocone.
Again, all triangular faces of the combined picture are commutative.


C
Φ
colim(D) = ♦
D



The commutativity condition on the cocone in essence forces interaction in the colimit of different parts of the diagram D.



The commutativity condition on the cocone in essence forces interaction in the colimit of different parts of the diagram D. Now stripping away the 'old' cocone gives the factorisation of the cocone via the colimit.

Memory Evolutive Systems

Memory Evolutive Systems

By A Ehresmann and J.P. Vanbremeersch, Elsevier, 2008

Memory Evolutive Systems

By A Ehresmann and J.P. Vanbremeersch, Elsevier, 2008

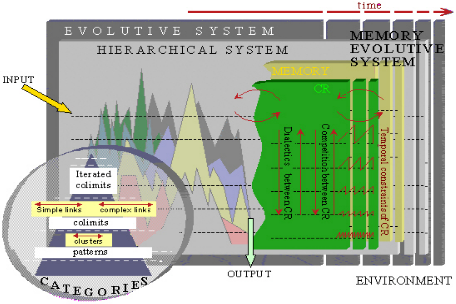
An MES is given by a category evolving over time so that the colimits change. They show that this concept can accommodate a wide range of problems considered important in biology, such as the binding problem.

Memory Evolutive Systems

By A Ehresmann and J.P. Vanbremeersch, Elsevier, 2008
An MES is given by a category evolving over time so that the colimits change. They show that this concept can accommodate a wide range of problems considered important in biology, such as the binding problem.
http://pagesperso-orange.fr/vbm-ehr/Ang/W209T.htm

Memory Evolutive Systems

By A Ehresmann and J.P. Vanbremeersch, Elsevier, 2008
An MES is given by a category evolving over time so that the colimits change. They show that this concept can accommodate a wide range of problems considered important in biology, such as the binding problem.
http://pagesperso-orange.fr/vbm-ehr/Ang/W209T.htm



"The theory of Memory Evolutive Systems represents a mathematical model for natural open self-organizing systems, such as biological, sociological or neural systems. In these systems, the dynamics are modulated by the cooperative and/or competitive interactions between the global system and a net of internal Centers of Regulation (CR) with a differential access to a central hierarchical Memory." (from the advertisement for the book).

"The theory of Memory Evolutive Systems represents a mathematical model for natural open self-organizing systems, such as biological, sociological or neural systems. In these systems, the dynamics are modulated by the cooperative and/or competitive interactions between the global system and a net of internal Centers of Regulation (CR) with a differential access to a central hierarchical Memory." (from the advertisement for the book).
DARPA Mathematical Challenges Defense Sciences Office;
DARPA-BAA 08-65 September 26, 2008
Mathematical Challenge One: The Mathematics of the Brain
Develop a mathematical theory to build a functional model of the brain that is mathematically consistent and predictive rather than merely biologically inspired.

Higher dimensional algebra

Higher dimensional algebra

The brain surely forms at the least a highly distributed, concurrent processing system.

Higher dimensional algebra

The brain surely forms at the least a highly distributed, concurrent processing system. We surely cannot imagine that it works entirely in serial processing.

Higher dimensional algebra

The brain surely forms at the least a highly distributed, concurrent processing system. We surely cannot imagine that it works entirely in serial processing. How can we imagine the higher dimensional processes which are used?

Higher dimensional algebra

The brain surely forms at the least a highly distributed, concurrent processing system. We surely cannot imagine that it works entirely in serial processing. How can we imagine the higher dimensional processes which are used?
Deeper results require higher dimensional rewriting!

Higher dimensional algebra

The brain surely forms at the least a highly distributed, concurrent processing system. We surely cannot imagine that it works entirely in serial processing. How can we imagine the higher dimensional processes which are used?

Deeper results require higher dimensional rewriting!

Local-to-global:

Higher dimensional algebra

The brain surely forms at the least a highly distributed, concurrent processing system. We surely cannot imagine that it works entirely in serial processing. How can we imagine the higher dimensional processes which are used?

Deeper results require higher dimensional rewriting!

Local-to-global: Use local rules to get global deductions.

Higher dimensional algebra

The brain surely forms at the least a highly distributed, concurrent processing system. We surely cannot imagine that it works entirely in serial processing. How can we imagine the higher dimensional processes which are used?
Deeper results require higher dimensional rewriting!
Local-to-global: Use local rules to get global deductions.
Not too difficult in dimension 2,

Higher dimensional algebra

The brain surely forms at the least a highly distributed, concurrent processing system. We surely cannot imagine that it works entirely in serial processing. How can we imagine the higher dimensional processes which are used?

Deeper results require higher dimensional rewriting!

Local-to-global: Use local rules to get global deductions.

Not too difficult in dimension 2,

hard in dimension 3, and

Higher dimensional algebra

The brain surely forms at the least a highly distributed, concurrent processing system. We surely cannot imagine that it works entirely in serial processing. How can we imagine the higher dimensional processes which are used?
Deeper results require higher dimensional rewriting!
Local-to-global: Use local rules to get global deductions.
Not too difficult in dimension 2,
hard in dimension 3, and
not yet done in dimension 4 (as far as I know).

Concurrency

Concurrency
Deadlock problem

Concurrency
Deadlock problem
Directed homotopy theory

Concurrency
Deadlock problem
Directed homotopy theory
No ‘fundamental groupoid’

Concurrency
Deadlock problem
Directed homotopy theory
No ‘fundamental groupoid’ instead, fundamental category.

Concurrency
Deadlock problem
Directed homotopy theory
No ‘fundamental groupoid’ instead, fundamental category.
Higher dimensional category theory gives a rich source of algebra and of concepts not yet integrated with analysis but with many applications to differential geometry.

Groupoid atlases:

Groupoid atlases: patching groupoids

Groupoid atlases: patching groupoids

The objects of a groupoid give a spatial component to group theory

Groupoid atlases: patching groupoids

The objects of a groupoid give a spatial component to group theory enabling a patching idea

Groupoid atlases: patching groupoids

The objects of a groupoid give a spatial component to group theory enabling a patching idea and a possibility of transferring from one patch to another.

Groupoid atlases: patching groupoids

The objects of a groupoid give a spatial component to group theory enabling a patching idea and a possibility of transferring from one patch to another.

Origins in algebraic K-theory:

Groupoid atlases: patching groupoids

The objects of a groupoid give a spatial component to group theory enabling a patching idea and a possibility of transferring from one patch to another.

Origins in algebraic K-theory: $GL(n, R)$ and its family of elementary subgroups,

Groupoid atlases: patching groupoids

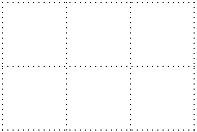
The objects of a groupoid give a spatial component to group theory enabling a patching idea and a possibility of transferring from one patch to another.

Origins in algebraic K-theory: $GL(n, R)$ and its family of elementary subgroups, with actions of the big group on cosets.

Need for a theory to accommodate for example:

Groupoid atlases: patching groupoids

The objects of a groupoid give a spatial component to group theory enabling a patching idea and a possibility of transferring from one patch to another.
Origins in algebraic K-theory: $GL(n, R)$ and its family of elementary subgroups, with actions of the big group on cosets.
Need for a theory to accommodate for example:



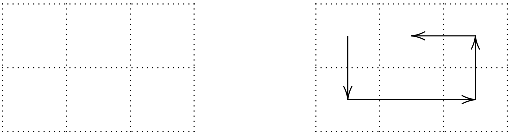
set with two
equivalence relations

Groupoid atlases: patching groupoids

The objects of a groupoid give a spatial component to group theory enabling a patching idea and a possibility of transferring from one patch to another.

Origins in algebraic K-theory: $GL(n, R)$ and its family of elementary subgroups, with actions of the big group on cosets.

Need for a theory to accommodate for example:



set with two equivalence relations

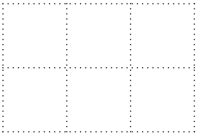
path through equivalence classes

Groupoid atlases: patching groupoids

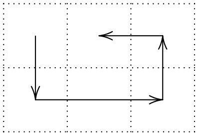
The objects of a groupoid give a spatial component to group theory enabling a patching idea and a possibility of transferring from one patch to another.

Origins in algebraic K-theory: $GL(n, R)$ and its family of elementary subgroups, with actions of the big group on cosets.

Need for a theory to accommodate for example:



set with two equivalence relations



path through equivalence classes

Similarly, a set with several group actions.

Aim is a homotopy theory of such objects with π_0, π_1, covering objects, . . .

Aim is a homotopy theory of such objects with π_0, π_1, covering objects, . . .
Groupoids include

Aim is a homotopy theory of such objects with π_0, π_1, covering objects, ...
Groupoids include equivalence relations,

Aim is a homotopy theory of such objects with π_0, π_1, covering objects, ...
Groupoids include equivalence relations, group actions,

Aim is a homotopy theory of such objects with π_0, π_1, covering objects, . . .
Groupoids include equivalence relations, group actions, bundles of groups,

Aim is a homotopy theory of such objects with π_0, π_1, covering objects, . . .

Groupoids include equivalence relations, group actions, bundles of groups, free groupoids on directed graphs, . . .

Aim is a homotopy theory of such objects with π_0, π_1, covering objects, . . .

Groupoids include equivalence relations, group actions, bundles of groups, free groupoids on directed graphs, . . .

A. Bak, Global actions: The algebraic counterpart of a topological space. *Uspeki Mat. Nauk., English Translation: Russian Math. Surveys* 525 (1997) 955 –996.

Aim is a homotopy theory of such objects with π_0, π_1, covering objects, ...

Groupoids include equivalence relations, group actions, bundles of groups, free groupoids on directed graphs, ...

A. Bak, Global actions: The algebraic counterpart of a topological space. *Uspeki Mat. Nauk., English Translation: Russian Math. Surveys* 525 (1997) 955 –996.

Group action of G on X corresponds to an action groupoid $G \ltimes X$.

Aim is a homotopy theory of such objects with π_0, π_1, covering objects, ...

Groupoids include equivalence relations, group actions, bundles of groups, free groupoids on directed graphs, ...

A. Bak, Global actions: The algebraic counterpart of a topological space. *Uspeki Mat. Nauk., English Translation: Russian Math. Surveys* 525 (1997) 955 –996.

Group action of G on X corresponds to an action groupoid $G \ltimes X$.

149. (with Bak, A., Minian, G., and Porter, T.), 'Global actions, groupoid atlases and applications', J. Homotopy and Related Structures, 1 (2006) 101-167.

Aim is a homotopy theory of such objects with π_0, π_1, covering objects, ...
Groupoids include equivalence relations, group actions, bundles of groups, free groupoids on directed graphs, ...
A. Bak, Global actions: The algebraic counterpart of a topological space. *Uspeki Mat. Nauk., English Translation: Russian Math. Surveys* 525 (1997) 955 –996.
Group action of G on X corresponds to an action groupoid $G \ltimes X$.
149. (with Bak, A., Minian, G., and Porter, T.), 'Global actions, groupoid atlases and applications', J. Homotopy and Related Structures, 1 (2006) 101-167.
Mathias del Hoyo, Gabriel Minian ' Classical Invariants for Global Actions and Groupoid Atlases', Applied Cat. Strs. (last December).

Data:

Data:
an indexing set Φ

Data:
an indexing set Φ
a reflexive relation $\leqslant$,

Data:
an indexing set Φ
a reflexive relation $\leqslant$,
a set X

Data:
an indexing set Φ
a reflexive relation $\leqslant$,
a set X
for each $\alpha \in \Phi$ a groupoid G_α,

Data:
an indexing set Φ
a reflexive relation $\leqslant$,
a set X
for each $\alpha \in \Phi$ a groupoid G_α, $Ob(G_\alpha) \subseteq X_\alpha$,

Data:
an indexing set Φ
a reflexive relation $\leqslant$,
a set X
for each $\alpha \in \Phi$ a groupoid G_α, $Ob(G_\alpha) \subseteq X_\alpha$,
$\alpha \leqslant \beta \Rightarrow$ morphism of groupoids

$$\phi_{\alpha,\beta} : G_\alpha|(X_\alpha \cap X_\beta) \to G_\beta|(X_\alpha \cap X_\beta)$$

Data:
an indexing set Φ
a reflexive relation $\leqslant$,
a set X
for each $\alpha \in \Phi$ a groupoid G_α, $Ob(G_\alpha) \subseteq X_\alpha$,
$\alpha \leqslant \beta \Rightarrow$ morphism of groupoids

$$\phi_{\alpha,\beta} : G_\alpha|(X_\alpha \cap X_\beta) \to G_\beta|(X_\alpha \cap X_\beta)$$

which restricts to the identity on objects.

Data:
an indexing set Φ
a reflexive relation $\leqslant$,
a set X
for each $\alpha \in \Phi$ a groupoid G_α, $Ob(G_\alpha) \subseteq X_\alpha$,
$\alpha \leqslant \beta \Rightarrow$ morphism of groupoids

$$\phi_{\alpha,\beta} : G_\alpha|(X_\alpha \cap X_\beta) \to G_\beta|(X_\alpha \cap X_\beta)$$

which restricts to the identity on objects.
The only axiom is that if $\alpha \leqslant \beta$ then $X_\alpha \cap X_\beta$ is a union of components of G_α,
i.e. if $x \in X_\alpha \cap X_\beta$ and $g \in G_\alpha$ is such that $s(g) = x$ then $t(g) \in X_\alpha \cap X_\beta$.

Data:
an indexing set Φ
a reflexive relation $\leqslant$,
a set X
for each $\alpha \in \Phi$ a groupoid G_α, $Ob(G_\alpha) \subseteq X_\alpha$,
$\alpha \leqslant \beta \Rightarrow$ morphism of groupoids

$$\phi_{\alpha,\beta} : G_\alpha|(X_\alpha \cap X_\beta) \to G_\beta|(X_\alpha \cap X_\beta)$$

which restricts to the identity on objects.
The only axiom is that if $\alpha \leqslant \beta$ then $X_\alpha \cap X_\beta$ is a union of components of G_α,
i.e. if $x \in X_\alpha \cap X_\beta$ and $g \in G_\alpha$ is such that $s(g) = x$ then $t(g) \in X_\alpha \cap X_\beta$.
Thus the fact that a groupoid has objects enables a patching idea and a possibility of transferring from one patch to another.

Mathematics as an evolving hierarchical system

Mathematics as an evolving hierarchical system

Communication between neurological structures is symbolic, or

Mathematics as an evolving hierarchical system

Communication between neurological structures is symbolic, or abstract.

Mathematics as an evolving hierarchical system

Communication between neurological structures is symbolic, or abstract.

Can we abstract from the progress of mathematics over the centuries to the modelling of hierarchical biological systems?

Mathematics as an evolving hierarchical system

Communication between neurological structures is symbolic, or abstract.

Can we abstract from the progress of mathematics over the centuries to the modelling of hierarchical biological systems?

Can we model in computers the way mathematicians actually work?

Concepts in mathematics, often called types, satisfy in common usage:

Concepts in mathematics, often called types, satisfy in common usage:

1. Inheritance

Concepts in mathematics, often called types, satisfy in common usage:

1. Inheritance This means that a concept such as associative binary operation has to be set up only once, and then is used, with its notion of representation, and its algorithms, in all other cases where it applies, as for example at least twice for addition and multiplication in the concept, or type, of ring.

Concepts in mathematics, often called types, satisfy in common usage:

1. Inheritance This means that a concept such as associative binary operation has to be set up only once, and then is used, with its notion of representation, and its algorithms, in all other cases where it applies, as for example at least twice for addition and multiplication in the concept, or type, of ring.

2. Types as first class variables

Concepts in mathematics, often called types, satisfy in common usage:

1. Inheritance This means that a concept such as associative binary operation has to be set up only once, and then is used, with its notion of representation, and its algorithms, in all other cases where it applies, as for example at least twice for addition and multiplication in the concept, or type, of ring.

2. Types as first class variables This means that for example the type of ring can be used in setting up the type of polynomial ring.

Concepts in mathematics, often called types, satisfy in common usage:

1. Inheritance This means that a concept such as associative binary operation has to be set up only once, and then is used, with its notion of representation, and its algorithms, in all other cases where it applies, as for example at least twice for addition and multiplication in the concept, or type, of ring.
2. Types as first class variables This means that for example the type of ring can be used in setting up the type of polynomial ring.
3. Coercion

Concepts in mathematics, often called types, satisfy in common usage:

1. Inheritance This means that a concept such as associative binary operation has to be set up only once, and then is used, with its notion of representation, and its algorithms, in all other cases where it applies, as for example at least twice for addition and multiplication in the concept, or type, of ring.
2. Types as first class variables This means that for example the type of ring can be used in setting up the type of polynomial ring.
3. Coercion Example: we need to be able to shift from a square matrix of polynomials such as

Concepts in mathematics, often called types, satisfy in common usage:

1. Inheritance This means that a concept such as associative binary operation has to be set up only once, and then is used, with its notion of representation, and its algorithms, in all other cases where it applies, as for example at least twice for addition and multiplication in the concept, or type, of ring.

2. Types as first class variables This means that for example the type of ring can be used in setting up the type of polynomial ring.

3. Coercion Example: we need to be able to shift from a square matrix of polynomials such as

$$\begin{bmatrix} 1+t & 1-t^2 \\ 2-t & 1+3t^2 \end{bmatrix}$$

Concepts in mathematics, often called types, satisfy in common usage:

1. Inheritance This means that a concept such as associative binary operation has to be set up only once, and then is used, with its notion of representation, and its algorithms, in all other cases where it applies, as for example at least twice for addition and multiplication in the concept, or type, of ring.

2. Types as first class variables This means that for example the type of ring can be used in setting up the type of polynomial ring.

3. Coercion Example: we need to be able to shift from a square matrix of polynomials such as

$$\begin{bmatrix} 1+t & 1-t^2 \\ 2-t & 1+3t^2 \end{bmatrix}$$

to the polynomial over the ring of square matrices

$$\begin{bmatrix} 1 & 1 \\ 2 & 1 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ -1 & 0 \end{bmatrix} t + \begin{bmatrix} 0 & -1 \\ 0 & 3 \end{bmatrix} t^2$$

Suggestion

Suggestion

These are characteristics of symbolic communication and so should be built into computer programs modelling mathematics,

Suggestion

These are characteristics of symbolic communication and so should be built into computer programs modelling mathematics, and into models of systems communication.

Suggestion

These are characteristics of symbolic communication and so should be built into computer programs modelling mathematics, and into models of systems communication. The cricket example shows there is a long way to go!