Groupoidification

John Baez
joint with James Dolan, Todd Trimble,
Alex Hoffnung, and Christopher Walker

Department of Mathematics
University of California, Riverside

January 7, 2009

James Dolan invented **degroupoidification**, which turns:

- groupoids into vector spaces;
- ‘spans’ of groupoids into linear operators.

James Dolan invented **degroupoidification**, which turns:

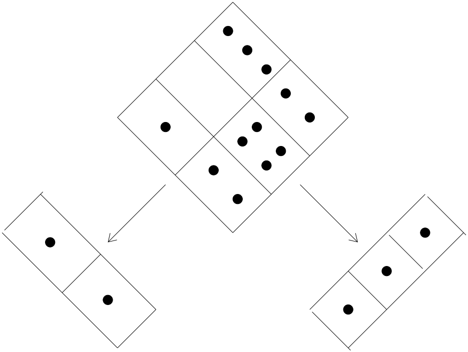
- groupoids into vector spaces;
- ‘spans’ of groupoids into linear operators.

A **span** from the groupoid X to the groupoid Y is a diagram

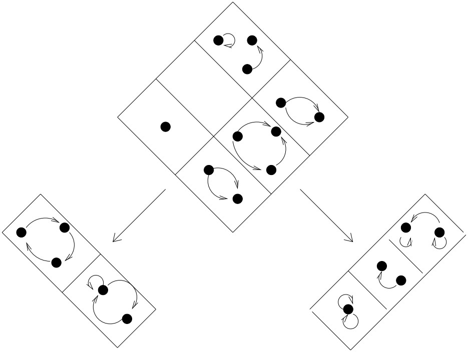
$$Y \xleftarrow{\;q\;} S \xrightarrow{\;p\;} X$$

where S is another groupoid, and p and q are functors.

A span of finite sets gives a matrix of natural numbers:



Using 'groupoid cardinality', a well-behaved span of groupoids gives a matrix of nonnegative real numbers:



We define the **cardinality** of a groupoid X to be:

$$|X| = \sum_{[x]} \frac{1}{|\mathrm{Aut}(x)|}$$

Here $[x]$ ranges over all isomorphism classes of objects in X.
$|\mathrm{Aut}(x)|$ is the order of the automorphism group of $x \in X$.

When this sum converges, we call X **tame**.

We define the **cardinality** of a groupoid X to be:

$$|X| = \sum_{[x]} \frac{1}{|\mathrm{Aut}(x)|}$$

Here $[x]$ ranges over all isomorphism classes of objects in X. $|\mathrm{Aut}(x)|$ is the order of the automorphism group of $x \in X$.

When this sum converges, we call X **tame**.

For example, the groupoid of finite sets has cardinality

$$\sum_{n=0}^{\infty} \frac{1}{|S_n|} \; = \; \sum_{n=0}^{\infty} \frac{1}{n!} \; = \; e$$

So: a sufficiently well-behaved span of groupoids

$$Y \xleftarrow{\;q\;} S \xrightarrow{\;p\;} X$$

can be viewed as a matrix of tame groupoids — and then turned into a matrix of nonnegative real numbers, $\underset{\sim}{S}$, using groupoid cardinality.

So: a sufficiently well-behaved span of groupoids

$$Y \xleftarrow{\;q\;} S \xrightarrow{\;p\;} X$$

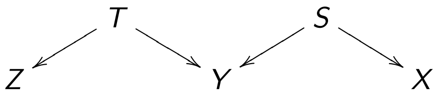
can be viewed as a matrix of tame groupoids — and then turned into a matrix of nonnegative real numbers, $\underset{\sim}{S}$, using groupoid cardinality.

BUT: the really good recipe for doing this involves a fudge factor you might not expect! We need this to get

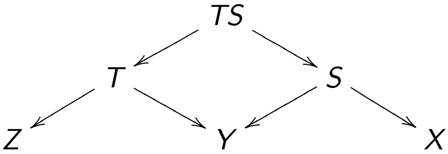
$$\underset{\sim}{TS} = \underset{\sim}{T}\,\underset{\sim}{S}$$

where TS is the composite of two spans.

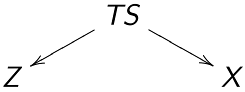
We compose spans of groupoids using 'weak pullback'. Given spans



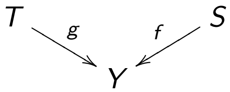
we can form a weak pullback in the middle:



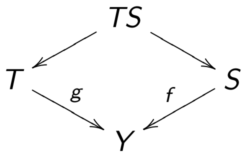
and get the **composite** span:



Given functors between groupoids



we define their **weak pullback** to be



where TS is the groupoid whose objects are triples consisting of $s \in S$, $t \in T$ and $\alpha \colon f(s) \xrightarrow{\sim} g(t)$.

Theorem

Any groupoid X gives a vector space called its **degroupoidification**:

$$\underset{\sim}{X} = \mathbb{C}^{\underline{X}}$$

where $\underline{X}$ is the set of isomorphism classes of objects in X. Any 'tame' span of groupoids

$$Y \xleftarrow{\;q\;} S \xrightarrow{\;p\;} X$$

gives a linear operator called its **degroupoidification**:

$$\underset{\sim}{S} \colon \underset{\sim}{X} \to \underset{\sim}{Y}$$

in such a way that

$$\underset{\sim}{TS} = \underset{\sim}{T}\,\underset{\sim}{S} \qquad\qquad \underset{\sim}{1_X} = 1_{\underset{\sim}{X}}$$

So, degroupoidification is a systematic process.

So, degroupoidification is a systematic process.

It's really a functor from the tricategory of:

- groupoids,
- tame spans,
- maps of spans,
- maps of maps of spans.

to the category of:

- vector spaces,
- linear operators.

Groupoidification is an attempt to reverse this process.

As with any form of categorification, this ‘reverse’ is not systematic. The idea is to take interesting pieces of linear algebra and reveal their combinatorial origin.

What can we groupoidify so far?

We can groupoidify the space of states of a quantum harmonic oscillator:

$$\mathbb{C}[[z_1, \ldots, z_n]] \;\cong\; \underset{\sim}{E^n}$$

where E^n is the groupoid of n-tuples of finite sets.

We can groupoidify the space of states of a quantum harmonic oscillator:

$$\mathbb{C}[[z_1, \ldots, z_n]] \;\cong\; \underset{\sim}{E^n}$$

where E^n is the groupoid of n-tuples of finite sets.

This lets us groupoidify:

- annihilation and creation operators:

$$a_i = \frac{\partial}{\partial z_i} \qquad a_i^* = \text{multiplication by } z_i$$

We can groupoidify the space of states of a quantum harmonic oscillator:

$$\mathbb{C}[[z_1, \ldots, z_n]] \;\cong\; \underset{\sim}{E}^n$$

where E^n is the groupoid of n-tuples of finite sets.

This lets us groupoidify:

- annihilation and creation operators:

$$a_i = \frac{\partial}{\partial z_i} \qquad a_i^* = \text{multiplication by } z_i$$

- field operators

$$\phi_i = a_i + a_i^*$$

  and their normal-ordered powers

We can groupoidify the space of states of a quantum harmonic oscillator:

$$\mathbb{C}[[z_1, \ldots, z_n]] \;\cong\; \underset{\sim}{E}^n$$

where E^n is the groupoid of n-tuples of finite sets.

This lets us groupoidify:

- annihilation and creation operators:

$$a_i = \frac{\partial}{\partial z_i} \qquad a_i^* = \text{multiplication by } z_i$$

- field operators

$$\phi_i = a_i + a_i^*$$

  and their normal-ordered powers
- the whole machinery of Feynman diagrams!

For any simply-laced Dynkin diagram D, we can groupoidify the q-deformed Borel subalgebra $U_q\mathfrak{b}$ when q is a prime power:

$$U_q\mathfrak{b} \;\cong\; \underset{\sim}{\mathrm{Rep}}(Q)$$

Here Q is a quiver corresponding to D, and $\mathrm{Rep}(Q)$ is the groupoid of representations of Q on finite-dimensional $\mathbb{F}_q$-vector spaces.

This is based on Ringel's work on Hall algebras.

For *any* Dynkin diagram D, we can groupoidify the Hecke algebra $H(D, q)$ when q is a prime power:

$$H(D, q) \;\cong\; (X \times X)//G$$

Here G is the simple algebraic group over $\mathbb{F}_q$ corresponding to D. Choosing a Borel subgroup $B \subset G$, we obtain the complete flag variety $X = G/B$.

$(X \times X)//G$ is the 'weak quotient' of $X \times X$ by G: a groupoid where two pairs of flags become *isomorphic* when there is an element of G mapping one to the other.

This is the beginning of a long story. For more, type

Groupoidification Made Easy

into Google or the arXiv.

This is the beginning of a long story. For more, type

Groupoidification Made Easy

into Google or the arXiv.

Also: listen to Alex Hoffnung's talk, coming up next!