Untyped linear $\lambda$-calculus and the combinatorics of 3-valent graphs

Noam Zeilberger (Ecole Polytechnique)

Combinatorics & Arithmetic for Physics: special days
2-3 December 2020 @ IHES (virtually!)
1. What is a "map"?
   (And how many are there?)
Topological definition

**map** = 2-cell embedding of a graph into a surface

considered up to deformation of the underlying surface.

*All surfaces are assumed to be connected and oriented throughout this talk*
Algebraic definition

**map** = transitive permutation representation of the group

\[ G = \langle v, e, f \mid e^2 = vef = 1 \rangle \]

considered up to \( G \)-equivariant isomorphism.

\[ v = (1\ 2\ 3)(4\ 5\ 6)(7\ 8\ 9)(10\ 11\ 12) \]
\[ e = (1\ 8)(2\ 11)(3\ 4)(5\ 12)(6\ 7)(9\ 10) \]
\[ f = (1\ 7\ 5\ 11)(2\ 10\ 8\ 3\ 6\ 9\ 12\ 4) \]

\[ c(v) - c(e) + c(f) = 2 - 2g \]
Combinatorial definition

**map** = connected graph + cyclic ordering of the half-edges around each vertex (say, as given by a planar drawing with "virtual crossings").
Graph versus Map
Some special kinds of maps

*planar*

*bridgeless*

*3-valent*
Four Color Theorem

The 4CT is a statement about maps.

every bridgeless planar map has a proper face 4-coloring

By a well-known reduction (Tait 1880), 4CT is equivalent to a statement about 3-valent maps

every bridgeless planar 3-valent map has a proper edge 3-coloring
Map enumeration

From time to time in a graph-theoretical career one's thoughts turn to the Four Colour Problem. It occurred to me once that it might be possible to get results of interest in the theory of map-colourings without actually solving the Problem. For example, it might be possible to find the average number of colourings on vertices, for planar triangulations of a given size.

One would determine the number of triangulations of 2n faces, and then the number of 4-coloured triangulations of 2n faces. Then one would divide the second number by the first to get the required average. I gathered that this sort of retreat from a difficult problem to a related average was not unknown in other branches of Mathematics, and that it was particularly common in Number Theory.

W. T. Tutte, Graph Theory as I Have Known It
Map enumeration

Tutte wrote a germinal series of papers (1962-1969)


One of his insights was to consider **rooted** maps

Key property: rooted maps have no non-trivial automorphisms
Map enumeration

Ultimately, Tutte obtained some remarkably simple formulas for counting different families of rooted planar maps, e.g.:

The number $a_n$ of rooted maps with $n$ edges is

$$\frac{2(2n)!3^n}{n!(n+2)!}.$$

For more on map-counting see:

Mireille Bousquet-Mélou, *Enumerative Combinatorics of Maps* (recorded lecture series)

Gilles Schaeffer, "Planar maps", in Handbook of Enumerative Combinatorics (ed. Bóna)

2. A crash course in (linear) $\lambda$-calculus
Lambda calculus: a very brief history*

Invented by Alonzo Church in late 20s, published in 1932

Original goal: foundation for logic without free variables

Minor defect: inconsistent!

Resolution: separate into an untyped calculus for computation, and a typed calculus for logic.

(Both have since found many uses.)

*Source: Cardone & Hindley's "History of Lambda-calculus and Combinatory Logic"
Turing published first *fixed-point combinator* (1937)
(key to Turing-completeness of $\lambda$-calculus)

$$(\lambda x.\lambda y.y(xxy))(\lambda x.\lambda y.y(xxy))$$

Observe *doubled uses* of variables $x$ and $y$.

By restricting to terms where every variable is used exactly once, one gets a well-behaved **linear** subsystem of lambda calculus.

(no longer Turing-complete...actually P-complete)
Untyped linear lambda terms (defn.)

basic judgment \( x_1, \ldots, x_n \vdash t \)

\( t \) is a linear term with free variables \( x_1, \ldots, x_n \)

inductive definition

\[
\frac{}{x \vdash x} \quad \frac{\Gamma \vdash t \quad \Delta \vdash u}{\Gamma, \Delta \vdash t(u)} \quad \frac{\Gamma \vdash \lambda x.t}{\Gamma, x \vdash t} \quad \frac{\Gamma, x, y, \Delta \vdash t}{\Gamma, y, x, \Delta \vdash t}
\]

define: subterms, bound variables, \( \alpha \)-equivalence, closed subterms, ordered terms
Untyped linear lambda terms (ex.)

$\vdash \lambda x.\lambda y.\lambda z.x(yz)$  \hspace{2cm} \text{ordered term (B)}

$\vdash \lambda x.\lambda y.\lambda z.(xz)y$  \hspace{2cm} \text{non-ordered term (C)}

$x \vdash \lambda y.\lambda z.x(yz)$  \hspace{2cm} \text{open term}

$x \vdash x(\lambda y.y)$  \hspace{2cm} \text{term w/closed subterm}
Term rewriting

Computation through **the rule of β-reduction:**

\[(\lambda x. t)(u) \rightarrow^{\beta} t[u/x] \]

can apply to any matching subterm (confluent and strongly normalizing)

Sometimes paired with **the rule of η-expansion:**

\[ t \rightarrow^{\eta} \lambda x. t(x) \]

**Example:**

\[
(\lambda x. \lambda y. \lambda z. x(yz))(\lambda a. a)(t) \\
\rightarrow^{\beta} (\lambda y. \lambda z. (\lambda a. a)(yz))(t) \\
\rightarrow^{\beta} (\lambda y. \lambda z. yz)(t) \\
\rightarrow^{\beta} \lambda z. t(z) \eta\leftarrow t
\]
**Typing**

**Types**

\[ A, B ::= X, Y, \ldots \mid A \multimap B \]

**Basic Judgment**

\[ x_1 : A_1, \ldots, x_n : A_n \vdash t : B \]

*t is a proof that \( A_1, \ldots, A_n \) (linearly) entail \( B \)*

**Inductive Definition**

\[
\begin{align*}
\Gamma \vdash t : A \multimap B & \quad \Delta \vdash u : A \\
\Gamma, \Delta \vdash t(u) : B \\
\Gamma, x : A \vdash t : B \\
\end{align*}
\]

\[
\Gamma, x : A, y : B, \Delta \vdash t : C \\
\Gamma, y : B, x : A, \Delta \vdash t : C
\]

*typed linear terms modulo \( \beta\eta \) present the free sym. closed multicategory!*
3. how on earth are these topics related??
An innocent idea

In May 2014, I thought it could be fun* to count untyped closed β-normal ordered linear terms by size (#λs)...

*for reasons related to certain categorical models of typing, cf. Melliès & Zeilberger POPL 2015
\( \lambda x. x \)
\( \lambda x. x(\lambda y. y) \)

\( \lambda x. \lambda y. x(y) \)
\[ \lambda x. x(\lambda y. (\lambda z. z)) \]
\[ \lambda x. x(\lambda y. (\lambda z. y(z))) \]
\[ \lambda x. x(\lambda y. \lambda z. y(z)) \]
\[ \lambda x. x(\lambda y. (\lambda z. z)) \]
\[ \lambda x. x(\lambda y. (\lambda z. y(z))) \]
\[ \lambda x. x(\lambda y. (\lambda z. z)) \]
\[ \lambda x. x(\lambda y. (\lambda z. y(z))) \]
\[ \lambda x. x(\lambda y. (\lambda z. z)(y)) \]
\[ \lambda x. x(\lambda y. (\lambda z. z)) \]
\[ \lambda x. x(\lambda y. (\lambda z. z)) \]
\[ \lambda x. x(\lambda y. (\lambda z. y(z))) \]
\[ \lambda x. x(\lambda y. (\lambda z. y(z))) \]
\[ \lambda x. x(\lambda y. (\lambda z. y(z))) \]
\[ 9 \]
\[ \lambda x. \lambda y. y(\lambda z. (\lambda w. y(\lambda z. (\lambda w. z)(\lambda w. w)))(\lambda w. w)) \]

\[ \lambda x. \lambda y. y(\lambda z. (\lambda w. y(\lambda z. (\lambda w. z)(\lambda w. w)))(\lambda w. w)) \]

\[ \lambda x. \lambda y. y(\lambda z. (\lambda w. y(\lambda z. (\lambda w. z)(\lambda w. w)))(\lambda w. w)) \]

\[ \lambda x. \lambda y. y(\lambda z. (\lambda w. y(\lambda z. (\lambda w. z)(\lambda w. w)))(\lambda w. w)) \]
Sequence: A000168

2*3^n*(2*n)!/(n!*(n+2)!).

Number of rooted planar maps with n edges. - Don Knuth, Nov 24 2013
Number of rooted 4-regular planar maps with n vertices.
Also, number of doodles with n crossings, irrespective of the number of loops.
The number $a_n$ of rooted maps with $n$ edges is

$$\frac{2(2n)!3^n}{n!(n+2)!}.$$ 

Number of rooted planar maps with $n$ edges. - Don Knuth, Nov 29 2012

Number of rooted 4-regular planar maps with $n$ vertices.

Also, number of doodles with $n$ crossings, irrespective of the number of loops.
## One piece of a larger puzzle

<table>
<thead>
<tr>
<th>family of rooted maps</th>
<th>family of lambda terms</th>
<th>sequence</th>
<th>OEIS</th>
</tr>
</thead>
<tbody>
<tr>
<td>planar maps</td>
<td>normal ordered terms</td>
<td>1,2,9,54,378,2916,...</td>
<td>A000168</td>
</tr>
</tbody>
</table>

### One piece of a larger puzzle

<table>
<thead>
<tr>
<th>family of rooted maps</th>
<th>family of lambda terms</th>
<th>sequence</th>
<th>OEIS</th>
</tr>
</thead>
<tbody>
<tr>
<td>trivalent maps (genus $g \geq 0$)</td>
<td>linear terms</td>
<td>$1, 5, 60, 1105, 27120,...$</td>
<td>A062980</td>
</tr>
<tr>
<td>planar maps</td>
<td>normal ordered terms</td>
<td>$1, 2, 9, 54, 378, 2916,...$</td>
<td>A000168</td>
</tr>
</tbody>
</table>

O. Bodini, D. Gardy, A. Jacquot (2013), Asymptotics and random sampling for BCI and BCK lambda terms, TCS 502: 227-238.
<table>
<thead>
<tr>
<th>family of rooted maps</th>
<th>family of lambda terms</th>
<th>sequence</th>
<th>OEIS</th>
</tr>
</thead>
<tbody>
<tr>
<td>trivalent maps (genus g\geq0)</td>
<td>linear terms</td>
<td>1,5,60,1105,27120,...</td>
<td>A062980</td>
</tr>
<tr>
<td>planar trivalent maps</td>
<td>ordered terms</td>
<td>1,4,32,336,4096,...</td>
<td>A002005</td>
</tr>
<tr>
<td>bridgeless trivalent maps</td>
<td>unitless linear terms</td>
<td>1,2,20,352,8624,...</td>
<td>A267827</td>
</tr>
<tr>
<td>bridgeless planar trivalent maps</td>
<td>unitless ordered terms</td>
<td>1,1,4,24,176,1456,...</td>
<td>A000309</td>
</tr>
<tr>
<td>maps (genus g\geq0)</td>
<td>normal linear terms (mod \sim)</td>
<td>1,2,10,74,706,8162,...</td>
<td>A000698</td>
</tr>
<tr>
<td>planar maps</td>
<td>normal ordered terms</td>
<td>1,2,9,54,378,2916,...</td>
<td>A000168</td>
</tr>
<tr>
<td>bridgeless maps</td>
<td>normal unitless linear terms (mod \sim)</td>
<td>1,1,4,27,248,2830,...</td>
<td>A000699</td>
</tr>
<tr>
<td>bridgeless planar maps</td>
<td>normal unitless ordered terms</td>
<td>1,1,3,13,68,399,...</td>
<td>A000260</td>
</tr>
</tbody>
</table>

O. Bodini, D. Gardy, A. Jacquot (2013), Asymptotics and random sampling for BCI and BCK lambda terms, TCS 502: 227-238.
Z (2016), Linear lambda terms as invariants of rooted trivalent maps, J. Functional Programming 26(e21).
4. Between linear $\lambda$-terms and rooted 3-valent maps

(a bijection by Bodini et al 2013, as analyzed by Z 2016)
Idea (folklore*): representing $\lambda$-terms as graphs

Can represent a term as tree w/two kinds of nodes (@/$\lambda$), with "pointers" from $\lambda$-nodes to bound variables. This idea is especially natural for linear terms.

*The idea itself is natural and should probably be called folklore. The earliest explicit description I know of (currently) is in Knuth's "Examples of Formal Semantics" (1970), but it was developed more deeply and independently from different perspectives in the PhD theses of C. P. Wadsworth (1971) and R. Statman (1974).
\textbf{λ-graphs as string diagrams}

This idea can also be understood within the categorical framework of "string diagrams", by interpreting λ-terms (after D. Scott) as \textit{endomorphisms of a reflexive object}

\[
\begin{array}{ccc}
\rho : U & \xrightarrow{\eta} & U \\
\lambda & \xleftarrow{\beta} & U \\
\end{array}
\]

in a symmetric monoidal closed bicategory.
From linear $\lambda$-terms to rooted 3-valent maps

$\lambda x.\lambda y.\lambda z. x(yz)$

$\lambda x.\lambda y.\lambda z. (xz)y$

$x, y \vdash (xy)(\lambda z. z)$

$x, y \vdash x((\lambda z. z)y)$
From linear $\lambda$-terms to rooted 3-valent maps

$\lambda x. \lambda y. \lambda z. x(yz)$  
$\lambda x. \lambda y. \lambda z. (xz)y$  
$x, y \vdash (xy)(\lambda z. z)$  
$x, y \vdash x((\lambda z. z)y)$

(B)  
(C)
From rooted 3-valent maps to linear $\lambda$-terms

Step #1: generalize to 3-valent maps with $\partial$ of "free" edges, one marked as root.

Step #2: observe any such map must have one of the following forms:

- Disconnecting root vertex
- Connecting root vertex
- No root vertex
From rooted 3-valent maps to linear $\lambda$-terms

Step #3: observe this is exactly the inductive definition of linear $\lambda$-terms!
An example
An example
An example
An example

disconnecting
An example
An example

\[ \lambda a. \lambda b. \lambda c. \lambda d. \lambda e. a(\lambda f. c(e(b(df)))) \]
5. Coda

(From Lambda Calculus to the Four Color Theorem...and beyond?)
Typing as coloring

recall the typing rules:

\[
\begin{align*}
\frac{}{\Gamma \vdash x : A} & \quad \frac{\Gamma \vdash t : A \rightarrow B \quad \Delta \vdash u : A}{\Gamma, \Delta \vdash t(u) : B} & \frac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x. t : A \rightarrow B} \\
\frac{}{\Gamma, x : A, y : B, \Delta \vdash t : C} & \quad \frac{\Gamma, y : B, x : A, \Delta \vdash t : C}{\Gamma, x : A, y : B, \Delta \vdash t : C}
\end{align*}
\]

we can interpret types in any ab gp G, taking \( A \rightarrow B := B - A \).

claim: any ordered \( \lambda \)-term \( t \) has a typing in \( G = \mathbb{Z}_2 \times \mathbb{Z}_2 \), such that for every subterm \( u \) of \( t \), \( u \) has type \((0,0)\) iff \( u \) is closed.

challenge problem: find a direct proof!
Some tools for further exploration

George Kaye's \(\lambda\)-term visualiser and gallery

https://www.georgejkaye.com/lambda-visualiser/visualiser.html

https://www.georgejkaye.com/lambda-visualiser/gallery

Jason Reed's Interactive Lambda Maps Toy


LinLam: a library for experimental linear lambda calculus

https://github.com/noamz/linlam
Some tools for further exploration

George Kaye's $\lambda$-term visualiser and gallery
https://www.georgejkaye.com/lambda-visualiser/visualiser.html
https://www.georgejkaye.com/lambda-visualiser/gallery

Jason Reed's Interactive Lambda Maps Toy

LinLam: a library for experimental linear lambda calculus
https://github.com/noamz/linlam